2 of 2 DOCUMENTS

PATENT ABSTRACTS OF JAPAN

2000339098

GET EXEMPLARY DRAWING

December 8, 2000

STORAGE DOMAIN MANAGEMENT SYSTEM

INVENTOR: PANAS MICHAEL G; MERRELL ALAN R; ALTMAIER JOSEPH; LANE JERRY PARKER; TAYLOR JAMES A; PARKS RONALD L; TAYLOR ALASTAIR; NOLAN SHARI J; NESPOR JEFFERY S; HARRIS GEORGE W JR; RICHARD A RUGUOO JR

APPL-NO: 2000085205 (JP 12085205)

FILED: March 24, 2000

PRIORITY: March 25, 1999, 99 276428, United States of America (US); July 2, 1999, 99 346592, United States of America (US); July 2, 1999, 99 347042, United States of America (US); December 6, 1999, 99 455106, United States of America (US); January 12, 2000, 00 482213, United States of America (US)

ASSIGNEE: DELL USA LP, THE

INT-CL: G06F3/06, (Section G, Class 06, Sub-class F, Group 3, Sub-group 06); G06F12/00, (Section G, Class 06, Sub-class F, Group 12, Sub-group 00); G06F15/16, (Section G, Class 06, Sub-class F, Group 15, Sub-group 16)

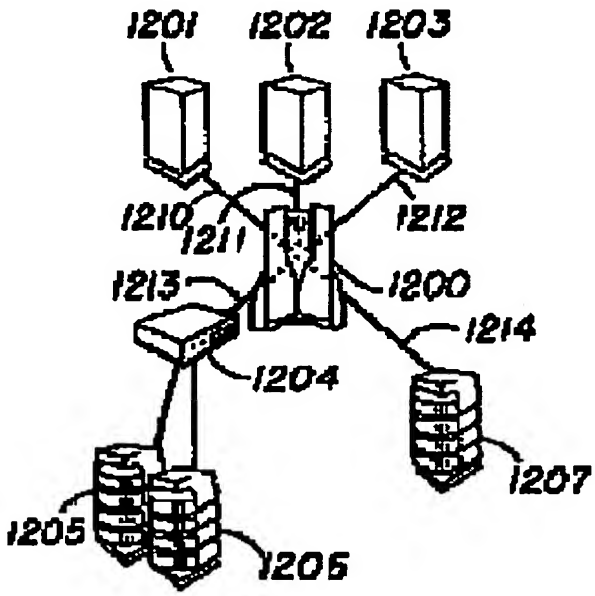
ABST:

PROBLEM TO BE SOLVED: To simplify the management of a storage system and also to effectively use both flexibility and capability of a storage area network(SAN) architecture by managing the storage resources in a storage network according to a storage domain.

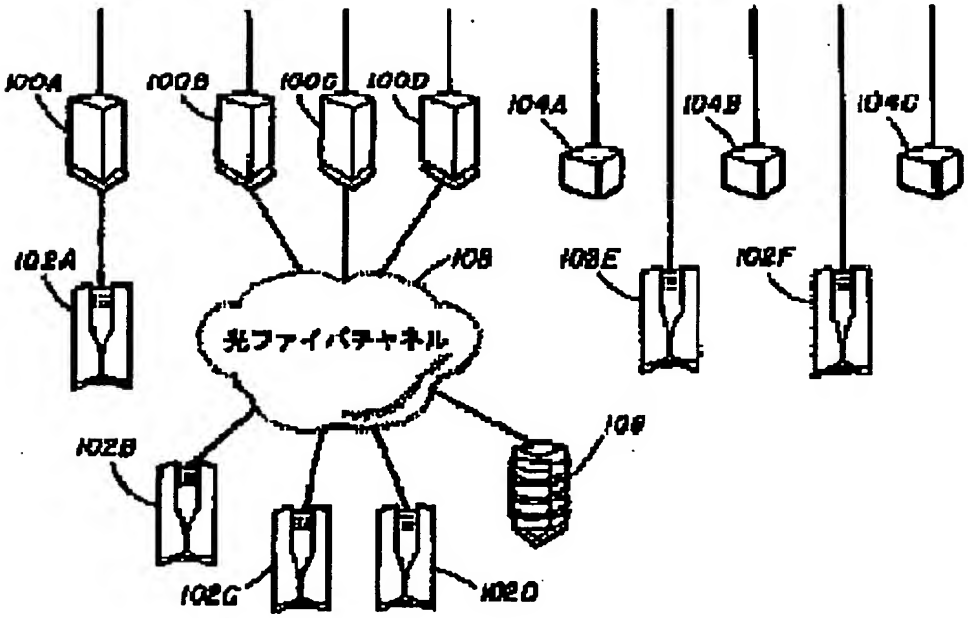
SOLUTION: A storage server 1200 in a network has the client interfaces 1210-1212 which are connected to the client servers 1201-1203 respectively. The storage interfaces 1213 and 1214 are connected to the storage devices 1205-1207 via the communication channels. These connected interfaces 1213 and 1214 are combined with some storages of the server 1200 and provide the physical storages for a storage domain which are managed in the server 1200. The server 1200 can induce a storage transaction by means of the local configuration data. Thus, the storage management is simplified for the client servers.

LOAD-DATE: July 11, 2001

Tips


1201
1202
1203
1210
1211
1212
1213
1200
1214
1204
1207
1205
1206

(a)


100A
100B
100C
100D
104A
104B
104C
102A
108
108E
102F
光ファイバチャネル
102B
106
102C
102D

(b)

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2000－339098
（P2000－339098A）
(43)公開日　平成12年12月８日(2000.12.8)

(51)Int.Cl.⁷	識別記号	ＦＩ	テーマコード（参考）
Ｇ０６Ｆ　3/06	３０１	Ｇ０６Ｆ　3/06	３０１Ａ
12/00	５４５	12/00	５４５Ａ
15/16	６４０	15/16	６４０Ｌ

審査請求　未請求　請求項の数18　ＯＬ　（全 38 頁）

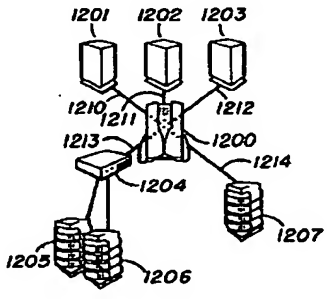
(21)出願番号　特願2000－85205(P2000－85205)
(22)出願日　平成12年３月24日(2000.3.24)
(31)優先権主張番号　２７６４２８
(32)優先日　平成11年３月25日(1999.3.25)
(33)優先権主張国　米国（ＵＳ）
(31)優先権主張番号　３４７０４２
(32)優先日　平成11年７月２日(1999.7.2)
(33)優先権主張国　米国（ＵＳ）
(31)優先権主張番号　３４６５９２
(32)優先日　平成11年７月２日(1999.7.2)
(33)優先権主張国　米国（ＵＳ）

(71)出願人　597001637
デル・ユーエスエイ・エルピー
ＤＥＬＬ　ＵＳＡ，Ｌ．Ｐ．
アメリカ合衆国テキサス州78682－2244，
ラウンド・ロック，ワン・デル・ウェイ
（番地なし）
(72)発明者　マイケル・ジー・バナズ
アメリカ合衆国、カリフォルニア州
94542、ハイワード、レオナ・ドライブ
24567
(74)代理人　100058479
弁理士　鈴江　武彦　（外４名）

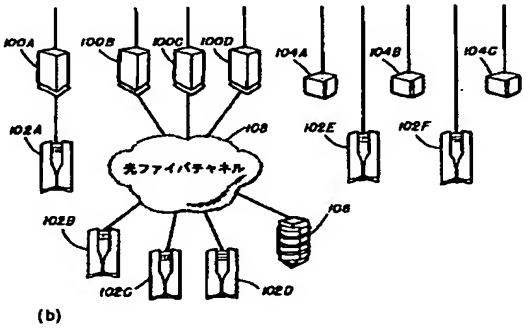
最終頁に続く

(54)【発明の名称】　ストレージドメイン管理システム

(57)【要約】
【課題】SANアーキテクチャのフレキイビリティ及び能力を活用しつつストレージシステムの管理を簡素化するシステムを提供する。
【解決手段】ストレージサーバは複数の通信インターフェースを有する。該インターフェースからなる第一のセットはあらゆる種類のデータユーザへの接続を担い、複数の通信インターフェースからなる第二のセットはストレージドメインで使用されるストレージデバイスプール内の各デバイスへの接続を担う。サーバ内のデータ処理リソースをこれら通信インターフェースに接続し、インターフェース間でデータ転送をする。データ処理リソースは複数のドライバモジュール及びこれらをデータパスに連結するよう構成可能なロジックからなる。構成された各データパスは前記複数のドライバモジュールから選択されたドライバモジュールのセットを含む仮想回路として動作する。



(a)



(b)

【特許請求の範囲】

【請求項１】　ストレージトランザクションが行われるクライアントの特定に十分な情報を伝えるそれぞれのストレージチャネルプロトコルを実行する一つ以上のクライアントおよび一つ以上のストレージシステムを含み、ストレージネットワークにおけるストレージドメインを管理するためのシステムにおいて、

前記一つ以上のクライアントおよび一つ以上のストレージシステムの各々一つと通信媒体を介して接続するために選定され、各種通信プロトコルに従って動作する複数の通信インターフェースと、

前記複数のインターフェースに結合され、前記一つ以上のストレージシステムからストレージロケーションセットを、前記一つ以上のクライアントからの少なくとも１つのクライアントセットのためのストレージドメインとして構成するロジックを含み、および特定されたクライアントに対応してストレージドメイン内でストレージトランザクションのルーティングを行うロジックを含む処理ユニットと、

前記複数の通信インターフェース間のストレージトランザクションを共通のフォーマットに変換し、あるいは該トランザクションから共通のフォーマットを変換によって得るロジックと、

不揮発性キャッシュメモリを含み、前記ストレージドメイン内の通信インターフェース間においてストレージトランザクションを共通フォーマットでルーティングする冗長リソースと、

前記処理ユニットに接続され、前記ストレージドメインを構成する管理インターフェースと、を具備することを特徴とするストレージドメイン管理システム。

【請求項２】　前記一つ以上のクライアントは、論理ストレージロケーションを特定するのに十分な情報を伝える各々のストレージチャネルプロトコルを実行し、前記論理ストレージロケーションに対応してストレージドメイン内でストレージトランザクションをルーティングするロジックを具備することを特徴とする請求項１に記載のストレージドメイン管理システム。

【請求項３】　ネットワークにおける１つのストレージロケーションから別のストレージロケーションへのデータセット移動を管理するロジックを具備することを特徴とする請求項１に記載のストレージドメイン管理システム。

【請求項４】　前記インターフェースがネットワーク内の複数のストレージドメインを構成するためのリソースを具備することを特徴とする請求項１に記載のストレージドメイン管理システム。

【請求項５】　ストレージネットワークにおけるストレージリソースの構成及び管理方法において、

ネットワーク内のクライアントおよびストレージリソース間に該ネットワークの中間システムをインストールし、

前記中間システムのロジックを用い、論理ストレージ範囲をネットワーク内のクライアントに割り当て、

前記中間システムのロジックを用い、ネットワーク内のストレージリソースを論理ストレージ範囲に割り当て、

前記クライアントに割り当てられた論理ストレージ範囲および前記論理ストレージ範囲に割り当てられたストレージリソースに従い、中間デバイスを通じてストレージトランザクションをルーティングすることを特徴とする方法。

【請求項６】　ストレージトランザクション通信チャネルをサポートする通信インターフェースと、

前記ストレージトランザクションチャネル上で受け取ったストレージトランザクションを内部フォーマットに変換するロジックと、

内部フォーマットのストレージトランザクションを、前記ストレージサーバとの通信における各々のデータストアとの接続を管理する仮想回路にルーティングするロジックと、を具備することを特徴とするストレージサーバ。

【請求項７】　仮想回路は、内部フォーマットを、対応する一つ以上のデータストアに関する一つ以上の通信プロトコルに変換するロジックを具備することを特徴とする請求項６に記載のストレージサーバ。

【請求項８】　対応する各々のデータソースに関する各々の通信プロトコルは、標準的な「インテリジェント入力／出力」(I_2O)メッセージフォーマットに適合するプロトコルを含むことを特徴とする請求項７に記載のストレージサーバ。

【請求項９】　仮想回路にストレージトランザクションをルーティングする前記ロジックはテーブルを含み、前記テーブルは複数のエントリを有し、前記複数のエントリは前記ストレージ通信チャネルで指定されたアドレス範囲と仮想回路の間の対応を示すことを特徴とする請求項６に記載のストレージサーバ。

【請求項１０】　仮想デバイスにストレージトランザクションをルーティングする前記ロジックはテーブルを含み、前記テーブルは複数のエントリを有し、前記複数のエントリは仮想回路と各データソースの間の対応を示すことを特徴とする請求項６に記載のストレージサーバ。

【請求項１１】　キャッシュを含み、仮想回路が前記キャッシュと通信することを特徴とする請求項６に記載のストレージサーバ。

【請求項１２】　各々のデータソースは不揮発性メモリを有することを特徴とする請求項６に記載のストレージサーバ。

【請求項１３】　各々のデータストアはハードディスクアレイを有することを特徴とする請求項６に記載のストレージサーバ。

【請求項１４】　コンフィギュレーションデータの入力

をサポートするユーザインターフェースを有することを特徴とする請求項６に記載のストレージサーバ。
【請求項１５】　前記ユーザインターフェースはグラフィカルユーザインターフェースからなることを特徴とする請求項１４に記載のストレージサーバ。
【請求項１６】　前記ユーザインターフェースは、前記ストレージサーバに接続されたタッチスクリーンからなることを特徴とする請求項１４に記載のストレージサーバ。
【請求項１７】　ストレージトランザクションのリクエストを発する少なくとも１つのクライアントシステムと、前記クライアントシステムに入り、及び該クライアントシステムから出る１つのクライアント通信チャネルと、複数のストレージデバイスと、複数のストレージデバイスに入り、及び該ストレージシステムから出る各々の通信チャネルと、を有するストレージネットワーク用サーバにおいて、
バスシステムを含むプロセッサと、
前記バスシステムに接続される前記クライアント通信チャネルへのクライアントインターフェースと、
前記バスシステムに接続される各々の通信チャネルへの複数のインターフェースと、
前記バスシステムに接続される不揮発性キャッシュメモリと、
前記サーバインターフェース上でストレージトランザクションのリクエストを受け取り、前記リクエストされたストレージトランザクションを前記複数のストレージデバイスに導き、前記ストレージトランザクションにおいて使用するよう前記不揮発性キャッシュメモリを割り当てるように前記プロセッサにより制御されるリソースと、を具備することを特徴とするサーバ。
【請求項１８】　前記プロセッサにより制御されるリソースは、ストレージトランザクションのためのアクセス許可を認証しおよび検証するプロセスを含むことを特徴とする請求項１７に記載のサーバ。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、大容量ストレージシステムの分野、特に、インテリジェントなストレージエリアネットワークのストレージトランザクション管理およびそのシステム構成に関する。
【０００２】
【従来の技術】いわゆる大容量記憶（ストレージ）システムに大量のデータを記憶させることは一般的となりつつある。大容量ストレージシステムは、通常、データネットワーク上のファイルサーバに連結されるストレージデバイスを含む。ネットワーク内のユーザはファイルサーバと通信してデータにアクセスする。ファイルサーバはデータチャネルを通じて特定のストレージデバイスに接続されているのが一般的であり、データチャネルには普通、ストレージトランザクション管理用に設計されたポイントツーポイント通信プロトコルが用いられる。
【０００３】記憶量と通信ネットワーク内のファイルサーバ数の増加に伴い、ストレージエリアネットワーク(SAN)の概念が提唱されてきた。ストレージエリアネットワークは、ストレージトランザクション用に最適化された通信ネットワーク内の多数の大容量ストレージシステムをつなぎ合わせたもので、例えば、光ファイバチャネルアービトレーティドループ(FC-AL)ネットワークはSANとして実装される。SANは、ストレージシステムのユーザとSAN上にある特定のストレージシステムとの間で行われるいくつものポイントツーポイント通信セッションをサポートする。
【０００４】ストレージシステムのファイルサーバおよび他のユーザは、特定の記録媒体と通信するよう構成されている。ストレージシステムを拡張したり、あるいはシステム内で媒体を交換すると、ファイルサーバおよび他のユーザにおいて再度構成が必要となる。また、いわゆるデータ移動作業にデータを１つの装置から別の装置に移す必要が生じると、移動プロセス中はそのデータへのアクセスをブロックしなければならないことが多い。また移動終了後は、ユーザシステムを再構成してからでなければ、新しいシステムからそのデータを利用することができない。
【０００５】概して、ストレージシステムとネットワークが複雑化してその規模が大きくなるにつれ、データのユーザ構成の管理、およびストレージシステム自体の構成管理についての問題が倍増する。
【０００６】
【発明が解決しようとする課題】本発明は上記事情を考慮してなされたものであり、ストレージシステムの管理を簡素化でき、その一方でSANアーキテクチャのフレキシビリティ及び能力を有効利用できるシステムおよび方法を提供することを目的とする。
【０００７】
【課題を解決するための手段】本発明はストレージドメイン管理のためのシステム、方法、およびサーバである。ここでストレージドメイン管理は、既存のストレージエリアネットワークハードウェアのインフラストラクチャ最上位に位置する中央集中的で安全な管理の能力を備えたものをいい、本発明は異機種混在型環境に適した高性能で信頼性の高い上級のストレージ管理を提供する。ストレージドメイン管理は、堅牢なストレージエリアネットワークファブリックの中枢として、新旧の機器を統合し、サーバおよびストレージリソースをネットワークとストレージ管理作業から解放する。また、ホストとしてネットワークベースのアプリケーションを処理し、ストレージエリアネットワークの全コンポーネントを通じてこれらのアプリケーションを活用できるようにする。ストレージドメイン管理によれば、従来のシステ

ムやテクニックではなし得なかった異機種混在型のストレージエリアネットワーク環境の構築、最適化が可能となる。

【０００８】本発明は、ストレージドメインに従ってストレージネットワーク内のストレージリソースを管理するためのシステムを提供する。このシステムには、通信媒体を通じてクライアントとストレージシステムおよびストレージネットワークに接続される複数の通信インターフェースが設置されている。これら複数の通信インターフェースには処理ユニットが接続され、この処理ユニットが有するロジックにより、ストレージネットワーク内にある一つ以上のクライアントのうち少なくとも１つのクライアントセットに対応するストレージドメインとして、同ネットワーク内の一つ以上のストレージシステムから１つのストレージロケーションセットが構成される。このシステムは、複数の通信インターフェースを通じたマルチプロトコルサポート、そのプロトコル内のトランザクション識別子に反応してストレージドメイン内でストレージトランザクションのルーティングを行うロジック、ストレージドメインを構成する管理インターフェース、複数の通信インターフェースにわたって実行されるストレージトランザクションを複数の通信インターフェースにあるシステム内でルーティングを行うための共通フォーマットに変換し、およびこの共通フォーマットから別のものに変えるためのロジック、ストレージトランザクションのデータサブジェクトを捕獲するリソース、ネットワーク内のあるストレージロケーションから別のストレージロケーションへのデータ移動を管理するロジックからなる、変形可能な種々の組み合わせ要素を含む。

【０００９】一実施の形態において、本発明のシステムはストレージエリアネットワーク内の中間デバイスとして、ファイルサーバ等のクライアントプロセッサと、クライアント用ストレージドメイン内のストレージリソースとして使用されるストレージシステムとの間に設置される。ストレージトランザクションは中間デバイスが受け取り、中間デバイスの構成ロジックによって定められるストレージドメインの構成に応じて管理される。中間デバイスは、ストレージエリアネッワーク内の管理サイトを提供し、これによってフレキシブルな構成、リダンダンシー、フェイルオーバ、データ移動、捕獲、複数プロトコルをサポートできる。さらに、一実施形態における中間デバイスはレガシーシステムエミュレーションを実行し、ストレージドメインにはクライアント用のレガシーストレージデバイスが含まれるため、クライアントの再構成が不要となる。

【００１０】ストレージドメインは、ネットワーク内のクライアントに論理ストレージ範囲を割り当て、ネットワーク内のストレージリソースをクライアントの論理ストレージ範囲にマッピングすることによって管理される。論理ストレージ範囲のクライアントへの割り当ては、中間システムあるいは、ネットワーク内のストレージリソースのクライアントから論理的に独立した、あるいは孤立したその他システムの中でクライアントに割り当てられた論理ストレージ範囲をマッピングすることによって完了する。このように、ストレージドメインマネジャを通じてアクセス可能なストレージリソースのストレージドメインは、ストレージドメインマネジャを中間デバイスとして使うことによって管理される。

【００１１】本発明によるストレージサーバは、処理ユニット、当該処理ユニットに接続されたバスシステム、通信インターフェース、当該処理ユニットに接続されたオペレーティングシステムからなる。バスシステムにはスロットがあり、これはこのスロットに接続されたサーバシャーシ上あるいは通信チャネル上のデータストアへのインタフェースを受けることができるようになっている。オペレーティングシステムはバスシステム上の転送を制御するロジックおよび通信インターフェース上でクライアントサーバから受け取るストレージトランザクションを内部フォーマットに変換するロジックのほか、コンフィギュレーションデータに応じて内部フォーマットを処理するロジックを提供し、このコンフィギュレーションデータはトランザクションプロトコル範囲で特定のストレージユニットに関する通信インタフェース上のストレージトランザクションを、内部フォーマットを使ってその範囲に対応する仮想回路にマッピングする。すると、仮想回路はインターフェース内の一つ以上のドライバを通じた一つ以上の物理データストアへのトランザクションのルーティングを管理する。また、サーバには物理的ストレージデバイスをエミュレートするためのリソースが含まれるため、クライアントサーバはストレージトランザクションのためにクライアントサーバの構成を変更せずに、仮想デバイスにアクセスするための標準的ストレージトランザクションプロトコルを使うことができる。

【００１２】本発明の別の要素によれば、ストレージルータが提供され、このストレージルータは第一の通信インタフェース、別の通信インタフェース、処理ユニットおよびバスシステムで構成される。バスシステムは処理ユニット、第一の通信インタフェース、別の通信インタフェースに接続されている。処理ユニットはオペレーティングシステムをサポートし、オペレーティングシステムは仮想デバイスのアーキテクチャとエミュレーションを使って、第一の通信インターフェース上で受け取ったストレージトランザクションをコンフィギュレーションデータに応じて別の適当な通信インターフェースに誘導する。

【００１３】いくつかの実施形態において、通信インタフェースは光ファイバ媒体へのインタフェースである。また、実施形態により、通信インタフェースが光ファイ

バチャネルアービトレーティドループに適合するドライバを含むものや、標準的な「小型計算機周辺機器インターフェース規格」バージョン3 (SCSI-3)に適合するドライバを含むものもある。

【００１４】いくつかの実施形態において、処理ユニットは複数の処理ユニットからなる。いくつかの実施形態において、バスシステムは相互接続されたコンピュータバスで構成され、実施形態によってはコンピュータバスが標準的な「周辺コンポーネント相互接続」(PCI)バスに適合するものもある。いくつかの実施形態において、通信インターフェースはバスシステムに接続される。

【００１５】いくつかの実施形態において、ストレージサーバは不揮発性メモリを有し、またいくつかの実施形態において、不揮発性メモリはフラッシュメモリ等の集積回路不揮発性メモリである。

【００１６】いくつかの実施形態において、ストレージサーバはディスクドライブ用コントローラを有し、いくつかの実施形態においてこのコンローラは標準的な「独立ディスクの冗長アレイ」(RAID)プロトコルをサポートする。いくつかの実施形態において、ディスクドライブは光ファイバ媒体によってコントローラと接続され、またいくつかの実施形態において、ディスクドライブは光ファイバ媒体に接続するためのデュアルインタフェースを有する。各ディスクドライブが少なくとも2つのコントローラに接続される実施形態もある。

【００１７】実施形態によっては、オペレーティングシステムは通信インターフェース上で受け取ったSCSI-3によるインストラクションとデータを内部フォーマットに変換するためのロジックを含むものや、SCSI-3インストラクションに対応する論理ユニット番号(LUN)を使って、SCSI-3インストラクションとデータがストレージサーバ内にデータストアを有する仮想デバイスに関連付けられるものもある。また、イニシエータSCSI-3識別番号(ID)とLUNを使って、SCSI-3インストラクションとデータがストレージサーバ内にデータストアを有する仮想デバイスに関連付けられる実施形態も可能である。

【００１８】いくつかの実施形態において、オペレーティングシステムはストレージサーバの動作とステータスをモニターするためのロジックを有し、また別の実施形態においては、デバイスの故障を扱い、コントロールを冗長コンポーネントに移行させるためのロジックがある。

【００１９】本発明は、データを記録、管理するための仮想デバイスと仮想回路をサポートするストレージサーバアーキテクチャを提供する。本発明によるストレージサーバには複数の通信インタフェースが搭載されており、これら複数の通信インタフェースにおける第一のセットはあらゆる種類のデータユーザに接続するためのもので、第二の通信インタフェースセットはストレージデバイス群の各デバイスに接続するためのものである。ストレージサーバのデータ処理リソースは複数の通信インタフェースに接続され、インタフェース間のデータ転送を可能にする。データ処理リソースは複数のドライバモジュールと、ドライバモジュールをデータパスにリンクさせる構成可能なロジックからなり、これらは好ましいシステムにおいてリダンダンシーを持たせるためにペアで実装される。構成されたデータパスはそれぞれ、複数のドライバモジュールから選択されたドライバモジュールセットを有する仮想回路の役割を果たす。通信インタフェースで受け取ったデータストレージトランザクションは、構成されたデータパスのひとつにマップされる。

【００２０】本発明の別の要素によれば、複数のドライバモジュールは複数の通信インタフェースにおける１つの通信インタフェースでサポートされるプロトコルのためのプロトコルサーバを有する。プロトコルサーバは、そのインタフェース上のプロトコルに従って特定のストレージ範囲を識別するターゲット識別子を認識する。特定のストレージ範囲にアドレスされたトランザクションは、サーバ内の特定の構成済みデータパスにマップされる。

【００２１】このように構成されたデータパスは仮想ストレージデバイスとして動作する。データのユーザは、特定のストレージデバイス用のプロトコルに従って、ストレージサーバ上の通信インタフェースと通信する。サーバ内では、そのプロトコルによるトランザクションがドライバセットによって実装される仮想ストレージデバイスにマップされる。特定のデータパスで実行されるストレージタスクのセットアップと変更および１つのデータパスから別のデータパスへのストレージ範囲マッピングのセットアップと変更は、ストレージサーバ内でドライバモジュールセットを構成することによって完了する。

【００２２】本発明の１つの要素によれば、複数のドライバモジュールは各通信インターフェースを管理する一つ以上のハードウェアドライブモジュールおよび複数の通信インターフェースとは独立してデータパスタスクを実行する一つ以上の内部ドライバモジュールを有する。データパスタスクには、例えばキャッシュメモリ管理、メモリミラーリング管理、メモリパーティション管理、データ移動管理、およびその他のストレージトランザクション管理タスクがある。仮想デバイスアーキテクチャでこの種のデータパスタスクを提供することにより、このようなタスクを管理するためのストレージシステムの構成は本質的にユーザにとってよくわかるものとなる。さらに、上記タスクを実行するように最適化されたストレージサーバに仮想デバイス機能を提供することで、性能改善とフレキシビリティの向上が実現する。

【００２３】また、本発明の１つの要素によれば、複数のドライバモジュールは、内部メッセージフォーマットに従ってサーバ環境内でデータを通信するためのロジッ

クを有する。受け取ったストレージトランザクションは内部メッセージフォーマットに変換され、特定のトランザクション用構成済みデータパスに入れられる。ある好ましい実施形態において、プロトコルサーバはプロトコル変換および仮想回路マッピングを実行する。

【００２４】構成可能なロジックには、コンフィギュレーションデータを受け入れるためのユーザインターフェースと、データパスから構成される各ドライバモジュールセットのテーブルまたはリストを記憶するメモリが含まれる。ひとつの実施形態における構成可能なロジックは、例えば入力信号を受け取るタッチスクリーンを有するディスプレイ上にグラフィカルユーザインターフェースを用いて実装される。グラフィカルユーザインターフェースにより、フレキシブルで使いやすい構成ツールを装備することができる。

【００２５】本発明の別の要素によれば、構成ロジックには、仮想回路用データパスを識別するテーブルの形態でコンフィギュレーションデータを記憶するメモリが含まれ、ある実施形態におけるこのメモリは、ストレージシステムのリセットや電源切断によってもデータが消失しない不揮発性メモリの中にテーブルを保持するパーシステントテーブルストレージプロセスを使って実現される。さらに、構成ロジックは、システム内の冗長ハードウェア上の冗長ドライバモジュールを使って仮想回路用のデータパスを実装しており、ストレージシステム上のどの故障箇所によっても特定のストレージトランザクションが妨げられることはない。好ましい実施形態において、ストレージドメイン内のリソースは、複数のドライバモジュールとドライバモジュールをデータパスに連結する構成可能なロジックからなる仮想回路を使って定義され、データパスは選好システムにリダンダンシーを持たせるためにペアで実装される。各構成済みデータパスは、複数のドライバモジュールから選択されたドライバモジュールセットを有する仮想回路の役割を果たす。通信インターフェースで受け取ったデータストレージトランザクションは構成されたデータパスのうちの１つにマップされ、こうして、ストレージドメインマネジャ内で管理、構成されるストレージドメイン内で管理される。

【００２６】基本的に、ストレージドメイン管理によって、ユーザはストレージエリアネットワークの機能を最大限に利用してビジネス上の問題に対処することができる。ストレージドメイン管理プラットフォームは各種ストレージシステムとプロトコルの異種間相互運用性、確実な中央集中的管理、スケーラビリティと優れた性能、信頼性、可用性、保守性といった特徴のすべてを、特定用途向けに作られたひとつのインテリジェントなプラットフォーム上で提供することができる。

【００２７】

【発明の実施の形態】以下、図面を参照しながら本発明の実施形態を説明する。

【００２８】図１（ａ）は、ストレージドメイン管理を行うインテリジェントなストレージエリアネットワーク(ISAN)サーバ1200を有するネットワークを示す。ストレージエリアネットワーク(SAN)は、クライアントコンピュータ用データストレージサービスを提供するために使用することができる。ストレージエリアネットワークは、ファイルサーバ、ウェブサーバ、エンドユーザコンピュータ等のクライアントコンピュータ用に広帯域、高スループットのストレージを提供するよう最適化されている。好ましい実施形態における本発明によるストレージサーバ1200は、シャーシ上でのデータストレージ、ストレージトランザクションキャッシュサービス、ストレージルーティングおよび仮想デバイス管理を可能にする。

【００２９】ネットワーク内のストレージサーバ1200は、クライアントサーバ1201，1202,1203にそれぞれ接続されたクライアントインタフェース1210，1211，1212を有する。ストレージインターフェース1213，1214は通信チャネルを通じてストレージデバイス1205，1206，1207に接続され、これらはストレージサーバ1200のいずれかのストレージと組み合わされると、ストレージサーバ1200内で管理されるストレージドメイン用の物理的ストレージを提供する。この例における通信チャネル1213は、ハブ1204を通じてデバイス1205，1206に接続されている。動作中、クライアントインターフェースは、クライアントサーバが例えば一つ以上のイニシエータ識別子、LUN番号等の論理範囲、ターゲットデバイスの識別子といったストレージドメインを識別できるパラメータを含むコマンドによってストレージトランザクションを要求するというプロトコルに従って動作する。ストレージサーバ1200は、リクエストされたトランザクションを仮想デバイスにマップし、この仮想デバイスが物理的ストレージデバイス間からのトランザクションに使用するよう物理的ストレージを割り当てる。ストレージサーバ1200はまた、リクエストの中で識別されたターゲットとなる物理的デバイスをエミュレートする資源を有する。ストレージサーバ1200は、ローカルコンフィギュレーションデータを使ってストレージトランザクションを誘導することができ、クライアントサーバ用のストレージ管理が簡略化される。

【００３０】スループットを最大限にするために、ストレージサーバ1200は光ファイバチャネルあるいはギガビットイーサネット等の高速ネットワーク媒体によって、クライアントサーバ1201-1203 に接続される。これらのクライアントサーバ1201-1203は、代表的な構成においては、ネットワークリンクによってエンドユーザコンピュータに接続される。

【００３１】図１（ａ）は、通信リンク109を通じてサーバ1200 に接続された管理インタフェース108を示す。ステーション108とサーバ1200内のインタフェースから

信号供給を受ける通信リンクは、各種実施形態において例えばイーサネットネットワークリンク、シリアルポートに接続されたシリアルケーブル、あるいはインターネットバスインターフェースで構成される。

【００３２】サーバ1201-1203とストレージデバイス1205-1207の間の通信は、中間デバイスとしてストレージサーバ1200を介し、光ファイバチャネルアービトレーティドループネットワークを通じて行われる。FC-AL上のチャネルは、好ましくは光ファイバチャネル媒体を使った標準的小型計算機および周辺機器インターフェース規格バージョン3 (SCSI-3)、別称、光ファイバチャネルプロトコル(FCP) (例えば、SCSI B X3T10, FCP X3.269-199X)に準じたプロトコルを使って実現される。別の実施形態においては、各種プロトコルでストレージトランザクションを実行する光ファイバチャネルファブリック上で、インターネットプロトコル等のプロトコルを使うこともできる。ストレージサーバ1200がデータストレージトランザクションの複数のプロトコルをサポートする実施形態もある。

【００３３】図1（ｂ）は、インテリジェントなストレージエリアネットワーク(ISAN)サーバのさまざまな用途を示す。ストレージエリアネットワーク(SAN)は、クライアントコンピュータ用のデータストレージサービスを提供するのに使用でき、ファイルサーバまたはウェブサーバ等のクライアントコンピュータ用に広帯域、高スループットのストレージを提供するよう最適化される。ISANサーバはデータの記録再生だけでなく、ストレージルーティング、仮想デバイス管理といった別の機能も提供する。

【００３４】図1（ｂ）は、サーバ100A-D, ISANサーバ102A-F, シンサーバ104A-C、ストレージアレイ106を含む。サーバ100A-Dは、UNIXサーバ、Windows™ NTサーバ、NetWare™サーバ、あるいはその他の種類のファイルサーバ、いずれでもよい。

【００３５】サーバ100A-Dは、ネットワークリンクによってクライアントコンピュータに接続される。ISANサーバ102Aは、ネットワークリンクによってサーバ100Aに連結され、リクエストされたストレージトランザクションを実行することによってサーバ100Aにデータストレージサービスを供給するため、サーバ100AはこのISANサーバ102Aをストレージデバイスのように扱う。ISANサーバ102Aは、代表的なハードディスクドライブまたはハードドライブアレイより多くのストレージを保有することができ、またストレージルータとして使用して、ISANサーバ102Aに接続されたデータストア間のインテリジェントなルーティングを提供することができる。

【００３６】ISANサーバ102Aはまた、一般的なハードディスクドライブやハードドライブアレイよりも広帯域でスループットの高いストレージトランザクション処理を行うことができるため、マルチメディアによるデータストリームおよびその他の大量データストリームが発する大量のデマンドを扱うことが可能である。

【００３７】最大限のスループットを得るために、ISANサーバ102Aは光ファイバチャネル等の高速ネットワーク媒体によってサーバ100Aに接続してもよい。サーバ100B-Dはネットワークリンクによってクライアントコンピュータに接続され、光ファイバチャネルファブリックによってストレージエリアネットワークに接続される。ストレージエリアネットワークはISANサーバ102B-Dとストレージアレイ106を含み、サーバ100B-DとISANサーバ102-B-Dは光ファイバチャネルアービトレーティドループ(FC-AL)用ドライバをサポートする。

【００３８】FC-AL上でのサーバ100B-Dとストレージデバイス間の通信は、好ましくは光ファイバチャネル媒体を使った標準的小型計算機および周辺機器インターフェース規格バージョン3 (SCSI-3)、別称、光ファイバチャネルプロトコル(FCP) (例えば、SCSI B X3T10, FCP X3.269-199X)に準じたプロトコルを使って実現される。別の実施形態においては、各種プロトコルでストレージトランザクションを実行する光ファイバチャネルファブリック108上で、インターネットプロトコル等のプロトコルを使うこともできる。ISANサーバ102Aが複数のプロトコルをサポートする実施形態もある。

【００３９】シンサーバ104A-Cはネットワークリンクを使ってクライアントに接続されるが、データストレージを提供するためのストレージエリアネットワークは使用しない。

【００４０】ISANサーバ102E-Fはネットワークリンクにより直接クライアントに接続され、中間ファイルサーバはない。ISANサーバ102E-Fは、ファイルサーバ、ウェブサーバその他の処理の機能を提供する特定用途向けプロセッサを提供することもできる。

【００４１】図2はストレージエリアネットワークの別の実施形態を示す。図２において、上記のようなストレージディレクタロジックとキャッシュメモリを備えたサーバ1250が各種プラットフォーム上のクライアントに接続されている。これらのプラットフォームは例えばHewlett-Packardサーバ1255、Sunサーバ1256、SGIサーバ1257であり、それぞれストレージトランザクション管理用に異なるプロトコルを実行する。ストレージドメインとして使われる物理的リソースを構成する複数の物理的ストレージデバイスもまたサーバ1250に接続され、上述の仮想デバイスアーキテクチャに従ってストレージディレクタによって管理される。この例における複数の物理的ストレージデバイスには、Hewlett-Packardプラットフォーム1251上のストレージ、Sunプラットフォーム1252上のストレージ、EMCプラットフォーム1253上のストレージがあ。このように、ストレージディレクタロジックを含むサーバは、従来のサーバおよびストレージをヘテロジニアス環境でサポートする共有ストレージプールを

作ることができる。複数のストレージデバイスおよびサーバ間の非互換性は、仮想デバイスアーキテクチャを使って、必要に応じてマスキングまたは模造することができる。こうして、真の意味でのストレージエリアネットワーク環境を利用し、ホスト、ファブリック、ストレージの相互運用性の問題をすべて、ストレージサーバレベルで管理することができる。

【００４２】仮想デバイスアーキテクチャを使ったストレージディレクタロジックは、ストレージドメイン構成を用いてクライアントサーバがストレージにアクセスする構成に関するひとつのインテリジェントな調整点を提供する。新たなデバイスを追加したり、既存のデバイスの管理を変える場合でも、ハードウェアの再構成はほとんど、あるいは全く不要となる。ストレージサーバの構成は、物理的ストレージにあるデータセットのサーバへのマッピングを自動的に保持することにより、正確な構成情報とコントロールを提供することができる。物理的ストレージを常に正確にマッピングすることは、ストレージエリアネットワークの管理を大幅に簡略化する。また、サーバのストレージディレクタにより、デバイスをオンライン状態にしたままで、古いストレージデバイスから新しいストレージデバイスへデータを移動させることができる。さらに、記録オブジェクトの大きさも、ひとつのアレイで作ることのできる最大オブジェクトのサイズによって制限されることがなくなる。複数アレイは、クライアントサーバ上で実行するホストオペレーティングシステムとは別に、ひとつのストレージオブジェクトに連結することができる。ストレージディレクタはまた、不揮発性キャッシュメモリ内のデータのスナップショットを作るといったバックアップおよびテスト動作を管理でき、例えばクライアントサーバを通じてルーティングすることなく、データをディスクからテープにコピーすることにより、データバックアップを管理することもできる。さらには、ローカルキャッシュを使って、ロストリダンダンシーを有するアレイからデータを移動し、アレイの修理、再構築中に冗長ストレージを修理し、データを十分に利用できる状態にすることが可能である。共通データセットにアクセスする複数サーバを有するアプリケーションでは、仮想デバイスアーキテクチャを使って拡張可能な単純なソリューションを提供するように、ストレージサーバ内にロッキングロジックを設置することができる。

【００４３】ストレージサーバ内のストレージディレクタロジックは、サーバとストレージ両方からのキャッシュ需要を統合するため、ストレージエリアネットワークに必要なキャッシュメモリ数が少なくて済む。このシステムは、クライアントサーバまたはストレージシステムのいずれにも、それぞれが内部メモリとして有効に提供できるものよりも多くのキャッシュを割り当てることができる。また、キャッシュはそのシステムを使うアプリケーション用の定義に合わせて、動的あるいは静的に割り当てられる。

【００４４】図３は、本発明による複数の相互接続されたストレージサーバを使った、より過密なストレージエリアネットワークの例を示す。ストレージサーバ1300，1301，1302は、通信チャネル1350，1351を使って相互接続され、例えば光ファイバチャネル、ギガビットイーサネット、非同期転送モード(ATM)等の高速プロトコルを使って搭載されている。本実施形態において、ストレージサーバはそれぞれストレージディレクタロジックと不揮発性キャッシュを有する。ストレージサーバ1300，1301，1302は、この例においては複数のクライアントサーバ1310 - 1318に接続され、クライアントサーバ1313と1314はハブ1320を通じてストレージサーバ1301に接続されている。同様に、クライアントサーバ1316 - 1318は、ハブ1321に接続され、ハブ1321はストレージサーバ1302に接続される。クライアントサーバ1310 - 1318は、先に詳述したFCP等のストレージチャネルプロトコルを用いてストレージサーバと通信する。これらのプロトコルによれば、ストレージトランザクションがリクエストされ、そのリクエストのイニシエータの識別子、論理ユニット番号(LUN)、そしてターゲットストレージデバイスの識別子が伝えられ、ストレージディレクタロジックがこれらのパラメータを使ってストレージドメイン内の仮想デバイスにストレージトランザクションをマップする。サーバにはまた、ターゲットストレージデバイスをエミュレートするためのリソースが含まれ、クライアントサーバはそのストレージエリアネットワーク内の複数のストレージデバイスとスムーズに相互運用できる。

【００４５】図３において、複数のストレージデバイス1330- 1339はストレージサーバ1300-1302 と接続されている。同図中の各種記号はストレージデバイスを示しており、ネットワークがヘテロジニアスで、サーバ1301、1302において仮想デバイスインターフェースによって管理されるさまざまなデバイスを利用できることがわかる。また、通信チャネルを変更することもできる。したがって、ハブ1340，1341，1342がネットワーク内に設けられ、ストレージデバイスとストレージサーバ間で各種の通信プロトコルが利用できるようになっている。［インテリジェントなストレージエリアネットワークサーバ］図４は、本発明によるストレージシステム管理リソースを含む、ある好ましい実施形態におけるストレージサーバのブロック図である。

【００４６】ストレージサーバ102は、ユーザおよびその他のデータ処理機能用の通信インターフェースセットを有する接続オプション130とストレージデバイス用の通信インターフェースセットを有するストレージオプション128とを有し、さらに、ハードウェアインターフェース126、オペレーティングシステム124、ブロックストレージインターフェース118、管理インターフェース12

0、プロトコルインターフェース122を有する。接続オプション130は、シリアル接続140；ある実施形態において構成管理ルーチンをサポートするフロントパネル接続142、遠隔管理ステーションとの通信をサポートするイーサネット接続144、ネットワークインターフェース145を有し、ストレージオプション128は、ドライブアレイ132、ソリッドステートドライブ(SSD)134、SCSIインターフェヘス136、ネットワークインターフェース138を有する。SCSIインターフェース136はDVD/CD-R 148に接続され、ネットワークインターフェース138はストレージサーバ102Gおよび/またはストレージ150に接続される。

【００４７】接続オプション130は、ストレージサーバにサーバとクライアントを接続する各種の方法である。シリアル接続140はネットワーク管理、遠隔管理用モデム、中断することのない電源供給メッセージを、フロントパネル接続142はストレージサーバ102のフロントパネルディスプレイとの管理接続を、またイーサネット接続144は管理プロトコルおよびおそらくはデータ転送用イーサネットインターフェースをそれぞれサポートする。ネットワークインターフェース146は、サーバ上に設置されるかもしれない多数の高速インターフェースのひとつである。ネットワークインターフェース146が光ファイバチャネルアービトレーティドループ(FC-AL)用ドライバを有する光ファイバチャネルインターフェースである実施形態もある。ネットワークインターフェース146には、光ファイバチャネルプロトコル(FCP)を使って光ファイバチャネル媒体上でのSCSI-3向けドライバを設けることも可能である。

【００４８】ハードウェアインターフェース126は、特定のハードウェアコンポーネントをインターフェースする。例えば、ネットワークインターフェース146は、構成、診断、動作モニター、健全さとステータスのモニターをサポートするための、特定のネットワークインターフェース向けソフトウェアモジュールセットを搭載する。

【００４９】オペレーティングシステム124、テーブル116、インターフェース118-122は、ストレージサーバ102の仮想デバイスとストレージルーティング機能をサポートする。ストレージサーバ102のこれらのコンポーネントは、システム内の構成されたドライバモジュールセットを使って、適当なストレージオプション128と接続オプション130間でストレージトランザクションのルーティングを行う。

【００５０】オペレーティングシステム124は、フェイルセーフ機能のほかに、メッセージのルーティング、転送機能も提供し、オペレーティングシステム124によるメッセージルーティング、転送機能は、ストレージサーバ102のコンポーネントの間でストレージトランザクション等のメッセージをルーティングするのに使用される。これらのメッセージには、仮想回路のコンポーネント間の内部フォーマットによるメッセージのほか、他のフォーマットによる制御メッセージが含まれる。

【００５１】ブロックストレージインターフェース118は、ブロックデータの転送をサポートするソフトウェアモジュールを提供する。インターフェース118は、ストライプ型データストレージ、ミラーリングされたデータストレージ、パーティションデータストレージ、メモリキャッシュのストレージ、RAIDストレージもサポートする。サポートされている各種のストレージタイプを連結し、例えばメモリキャッシュとミラーリングされたデータストレージ等、いろいろな組み合わせを作ることもできる。

【００５２】プロトコルインターフェース122は、さまざまなプロトコルによるリクエストを変換し、これに応えるソフトウェアモジュールを提供する。ひとつのモジュールセットは、イーサネット接続のレイヤに設置され、ハードウェアドライバ、データリンクドライバ、インターネットプロトコル(IP)ドライバ、転送制御プロトコル(TCP)ドライバ、ユーザデータグラムプロトコル(UDP)ドライバその他のドライバとなる。また別のモジュールセットはFCP用のドライバを提供する。

【００５３】管理インターフェース120は、ストレージサーバ102を管理するためのソフトウェアモジュールを提供し、テーブル116へのアクセスを管理するインターフェースのほか、スケジューリング、プロセス調整、システムのモニター、インフォームドコンセントの管理、システムプロセスやイベントの管理といった、ルールに基くシステム管理のためのインターフェースを含む。インフォームドコンセントの管理モジュールは、ストレージサーバ102を構成、維持するためのルールに基く管理策を講じておくことが前提となる。

【００５４】［ストレージトランザクションの操作］ストレージトランザクションは、接続オプション130のいずれかで受け取られる。ストレージトランザクションには、読み出し、書き込みリクエストおよびステータス問い合わせが含まれる。リクエストはブロック指向のものでもよい。

【００５５】典型的な読み出しストレージトランザクションは、読み出しコマンドとアドレシング情報で構成される。書き込みストレージトランザクションは、読み出しストレージトランザクションと似ているが、異なるのは、リクエストが送信されるデータ量に関する情報を含み、書き込むデータがこれに続く点である。より具体的には、SCSI-3プロトコルを使い、各デバイスは識別子(ID)を有する。リクエストを発生するマシンは、イニシエータと呼ばれ、リクエストに応えるマシンはターゲットと呼ばれる。この例において、サーバ100Aはイニシエータで ID7を持ち、ストレージサーバ102はターゲットで ID6を有する。SCSI-3プロトコルは2つ以上のアドレシングコンポーネント、論理ユニット番号(LUN)、アドレスを

提供する。

【００５６】LUNはターゲットIDのサブコンポーネントを特定する。例えば、複合型ハードディスク/テープドライブエンクロージャにおいて、2つのデバイスはひとつのIDを共有するかもしれないが、異なるLUNを有する。第三のアドレシングコンポーネントは、デバイスデータをどこから読み出し、どこに記憶するかというアドレスである。ストレージサーバ102Aはイニシエータごとのベースで仮想LUNを提供するため、ひとつのストレージサーバ102Aは、例えば1万以上の仮想LUNをサポートすることができる。

【００５７】ストレージサーバ102Aは、SCSI-3ストレージトランザクションリクエストを、ひとつの仮想LUNに対応する仮想回路にマップする。仮想回路は、一つ以上の仮想デバイスの連続である。仮想デバイスは、ソフトウェアモジュールまたはハードウェアコンポーネント等、一つ以上のデバイスで構成される。例えば、2つのネットワークインターフェースデバイスを組み合わせて仮想デバイスとしたり、同様に、2つのキャッシュデバイスを合わせて１つの仮想デバイスとすることができる。このデザインにより、コンポーネントが故障しても、ストレージサーバ102のストレージトランザクション処理機能に支障が生じることはない。

【００５８】仮想回路は、ストレージトランザクションをサポートするのに必要な仮想デバイスで構成される。通常、仮想回路内の第一のコンポーネントは、この例ではFCPであるストレージトランザクション通信チャネルフォーマットからのストレージトランザクションを内部フォーマットに変換するドライバである。このような内部フォーマットのひとつは、インテリジェントな入出力(I_2O)ブロックストレージアーキテクチャ(BSA)メッセージフォーマットと同様のものとすることができる。内部フォーマットは、好ましいシステムにおいて、ストレージ媒体と通信チャネルニュートラルである。

【００５９】仮想回路の中間仮想デバイスは、キャッシング、ミラーリング、RAIDといったその他の機能も供給する。内部フォーマットはストレージ媒体ニュートラルであるため、中間仮想デバイスは内部フォーマットで動作するように設計されており、同回路内の他の仮想デバイスと相互運用できる。

【００６０】仮想回路内の最後の仮想デバイスは通常、ストレージを管理するためのフォーマット変換および通信チャネルドライバである。例えば、ドライブアレイ132は、仮想デバイスを形成するようグループ分けされる冗長ハードウェアドライバモジュール(HDM)によって制御される。HDMはSCSI変換にBSAを供給し、HDMはドライブアレイ132を構成するドライブとのインターフェースを扱う。同様に、仮想回路がネットワークインターフェース138上の異なるストレージへのリンクである場合、ストレージデバイス通信チャネルプロトコルにBSAを変換するのをサポートする仮想デバイスが設けられる。

【００６１】ストレージサーバには、オペレーティングシステム内および物理的ストレージデバイスをエミュレートするクライアントサーバへのインターフェースでのリソースも含まれる。このエミュレーションにより、仮想デバイスはそのストレージにアクセスするクライアントサーバにとって、物理的デバイスであるかのように見える。したがって、クライアントサーバは、ストレージトランザクション用のSCSIコマンドを使って、FCP等の標準プロトコルによって通信するよう構成することができる。SCSIコマンドを利用する実施形態において、エミュレーションにはデバイス識別子を有するSCSIプロトコルと、開始サーバによって予測される、あるいはこれに適合するデバイス能力情報に応じて、問い合わせコマンドに対応することが関わる。また、SCSIプロトコルによる読み出し能力コマンドとモードページデータコマンドは、ストレージを用いるクライアントサーバが物理的ストレージデバイスに関する標準的構成情報に依存でき、その一方でストレージサーバが、クライアントサーバとのインターフェースで物理的ストレージデバイスをエミュレートすることによってクライアントサーバをスプーフし、実際のストレージトランザクションを仮想デバイスにマップするよう、エミュレーションリソースによって取扱われる。エミュレーションリソースにより、仮想デバイスはイニシエータ、論理ユニット番号(LUN)、ターゲットデバイス識別子との組み合わせによって識別することができ、ストレージトランザクションをリクエストにおいて識別された特定の物理的ターゲットデバイスに接続する必要はない。

【００６２】図5は、ストレージドメイン管理に用いるストレージ管理システム151として動作する、図4について示されているようなサーバの機能コンポーネントを示すブロック図である。システム151は、ストレージマネジャオペレーティングシステム152を有する。ストレージマネジャオペレーティングシステム152により、機能コンポーネントはストレージドメインルーティングリソース153、レガシーデバイスエミュレーションリソース154、データ移動リソース155、リダンダンシー、ホットスワップ及びフェイルオーバのリソース156を有する。ストレージマネジャオペレーティングシステムは、これらのリソース、オンシャーシキャッシュ157、管理インターフェース158、そして本実施形態においてはオンシャーシストレージアレイ159の通信を調整する。

【００６３】キャッシュ157は、本発明の一実施形態において、ストレージトランザクションを安全にサポートするためのソリッドステート不揮発性メモリアレイで構成される。別の実施形態において、キャッシュ157はさらに耐故障性を上げるために、冗長アレイを有する。

【００６４】システム151には、複数の通信インターフェース160-165が設置され、この例において、インター

フェース160はクライアントとストレージ管理システム151間のプロトコルXを、インターフェース161はクライアントとストレージ管理システム151の間のプロトコルYを、インターフェース162はストレージデバイスとストレージ管理システム151の間のプロトコルZを、インターフェース163はストレージデバイスとストレージ管理システムの間のプロトコルAを、インターフェース164はストレージデバイスとストレージ管理システム151の間のプロトコルBを、またインターフェース165はストレージマネジャシステム151とそのネットワーク上の他のストレージ管理システムとの間のプロトコルCを、それぞれ実行するように構成されている。

【００６５】図の例において、プロトコルX-ZおよびプロトコルA-Cは、ストレージ管理システム151によってサポートされており、これらのプロトコルは複数の異なるプロトコル、ひとつのプロトコルのバリエーション、あるいは、システムが利用される特定のストレージエリアネットワークに適したすべて同じプロトコルのいずれでもよい。

【００６６】ストレージトランザクションは、インターフェース160-165を通じて、それぞれの通信媒体からストレージ管理システム151の内部リソースへと行われる。好ましいシステムにおいて、ストレージトランザクションは、各種のインターフェースの中で、これらのインターフェースによって実行されるプロトコルとは独立してルーティングするための、システム内部の共通メッセージングフォーマットに変換される。ストレージドメインルーティングリソース153は、特定のクライアントデバイスとストレージデバイス用に構成された仮想回路を使って、ストレージドメイン内でトランザクションをマップする。レガシーエミュレーションリソース154とデータ移動リソース155により、新しい機器がネットワークに追加されたり、ネットワークから取り外される場合に、ストレージドメインはストレージ管理システム151において再構成される。例えば、新しいストレージデバイスをネットワークに追加することができ、既存のストレージデバイス内のデータセットを新しいストレージデバイスに移動でき、そのデータセットを使用したクライアントからのストレージトランザクションは、移動中およびターゲットエミュレーションを供給することによって移動が完了した後も、既存のストレージデバイスの中に残っているかのように見える。リダンダンシー、ホットスワップ、フェイルオーバリソース156により耐故障性が保たれ、高スループットのデータストレージネットワークでストレージ管理システム151が連続して動作できる。［ハードウェアアーキテクチャの概要］図６は、インテリジェントなストレージエリアネットワーク(ストレージ)サーバに適したハードウェアアーキテクチャの一例を示すブロック図である。ハードウェアアーキテクチャはリダンダンシーを利用し、分散型ソフトウェアシステムをサポートして、ひとつの故障箇所が特定のストレージトランザクションを妨害することがないようにしている。

【００６７】図６にはストレージサーバ102Aが搭載される。ストレージサーバは、標準的なコンポーネントと標準ベースのデバイスを使いながらも、高い冗長性を実現するよう設計されている。例えば、ストレージサーバ102Aは、標準的な周辺コンポーネント相互接続(PCIの高速バージョンと標準的光ファイバチャネルアービトレーティドループ(FC-AL)インターフェースを採用している。その他各種のプロトコルとインターフェースを使用する実施形態もある。

【００６８】ストレージサーバ102Aは、4つの分離した64ビット66メガヘルツPCIバス200A-Dを有する。ストレージデバイスとPCIバスのスロットにおけるネットワークインターフェースの構成は多数考えられる。一実施形態において、PCIバスは、SSD PCIバス200A-BとインターフェースPCIバス200C-Dの2グループに分けられ、各グループは上側、下側として指定される2つのバスを有する。各グループにおける上下のバスは冗長サービスを供給するように構成することができる。例えば、下側のSSD PCIバス200Bは上側のSSD PCI バス200A と同じ構成でもよい。

【００６９】PCIバス200A-Dはホストブリッジコントローラ(HBC)モジュール202A-Bに接続される。HBCモジュール202A-BはPCIバス200A-Dにまたがり、冗長ブリッジパスとなる。

【００７０】SSD PCI バス200A-Bは、ソリッドステートドライブ(SSD)モジュール204A-Gをサポートし、これらのSSDモジュール204A-Gは、フラッシュメモリストア等のソリッドステートストレージデバイスとなる。

【００７１】インターフェースPCIバスは、ネットワークインターフェースコントローラ(NIC)モジュール206A-B、独立ディスクの冗長アレイ(RAID)コントローラ(RAD)モジュール212A-B、および特定用途向け処理(ASP)モジュール208A-DからHBCモジュール202A-Bへの相互接続を行う。

【００７２】ストレージサーバ102Aを外部FC-ALに連結するのに加え、NIC 206A-Bは、光ファイバチャネルハブ(FCH)モジュール214A-Dに連結することができる。FCHモジュール214A-Dは各々、NICモジュール206A-Bの両方に接続され、FC-ALポートを10個ずつ提供し、NICモジュール206A-Bからカスケードされて20ステーションFC-ALハブを提供する。

【００７３】ディスクドライブハブ(DDH)モジュール216A-Dは冗長FC-ALファブリックを提供し、ディスクドライブをRACモジュール212A-Bに接続する。DDHモジュール216A-Dの各々におけるFC-ALファブリックは2つの冗長ループで構成され、これらはDDHモジュールに接続されたすべてのドライブをRACモジュール212A-Bの両方に連結す

る。RACモジュールはDHモジュール216A-D全部の間のループを管理し、DDHモジュール216A-Dはそれぞれディスクドライブ218等、5つのデュアルポートディスクドライブをサポートする。

【００７４】システムミッドプレーン(SMP)は図６に示されていない。SMPはパッシブミッドプレーンで、図６に示すように、HBCモジュール202A-B、SSDモジュール204A-H、RACモジュール212A-B、NICモジュール206A-B、FCHモジュール214A-D、DDHモジュール216A-D、ASPモジュール208A-Dの間の相互接続を実現する。SMPはコンパクトPCIベースで、4つのカスタムコンパクトPCIバス200A-D、RAC-DDH相互接続、NIC-FCH相互接続およびその他ミッドプレーン信号からなる制御バスを有する。さらに、SMPは電源サブシステム(図６では示されていない)からモジュールへ、電圧48V，12V，5V，3.3Vで配電する。

【００７５】フロントパネルディスプレイ(FPD)220は、ストレージサーバ102A用のユーザインターフェースを供給する。FPDにはディスプレイデバイスと入力デバイスが含まれ、一実施形態においては、タッチセンシティブの液晶ディスプレイ(LCD)を使って入力機能を有するタッチスクリーンとすることができる。FPD220はHBCモジュール202A-Bと接続され、ステータス表示、構成表示および管理、その他管理機能をサポートする。

【００７６】図６には示されていないが、電源およびファンサブシステムにより、冗長AC-DC電源供給、冗長DC-DC電源変換、電源停止用のバッテリーバックアップおよび冗長プッシュプルファンサブシステムが供給される。これらのコンポーネントは、ストレージエリアネットワークを活用する場合に重要となる高い可用性と低いダウンタイムという特徴をサポートする。

【００７７】ストレージサーバ102Aは他のストレージサーバと連結して、ストレージエリアネットワーク内のひとつのネットワークポートあるいはストレージデバイスに設置されたネットワークのようにすることができる。この接続は、HBCモジュール202A-Bの各々に接続されたFC-AL拡張ポート上で行われる。さらに、HBCモジュール202A-Bは帯域外管理のためにRS232シリアルポートと10/100イーサネットポートを提供する。

【００７８】バスシステムには、ストレージサーバ102A内のすべてのバスが含まれる。この例において、バスシステムはホストブリッジコントローラによって相互接続される4つのPCIバスを有し、また別のインターフェースを行うHBCモジュール内部のPCIバスも有する。スロットは、バスシステム上でインターフェースを受けられるすべての位置を含む。この例において、HBCモジュール外の4つのPCIバスはそれぞれ4つのインターフェースに対応することができる。

【００７９】インターフェースはカードあるいはスロットに入るその他のデバイスで、インターフェースに接続されるデータストア用のドライバおよびハードウェアをサポートする。[リダンダンシーとフェイルオーバ]ストレージサーバ102Aは高いリダンダンシーを提供する。一実施形態において、冗長NIC，RAC，HBCモジュールがある。SSDモジュールとドライブはミラーラングをサポートする。ドライブはまた、パリティおよびデュアルチャネルアクセスもサポートしている。DDHモジュールはそれぞれ、RACモジュールへの接続を行うための完全冗長FC-ALファブリックを含む。フェイルオーバはHBCモジュールが扱い、これはストレージサーバ内の他のモジュールを制御する。この制御はマルチレイヤ式である。

【００８０】HBCモジュールのコントロールオーバ第一レイヤは電源供給制御で、各モジュールは同モジュール上のCMBコントローラによって制御される個々の電源供給イネーブル信号を有する。HBCモジュールは冗長性があるものの、１つのHBCモジュールだけがマスタHCモジュールとして動作して、システムを誘導、制御し、他のHBCはスレーブとなる。モジュールをスロットにプラグ接続すると、その電源供給は当初ディスエーブルされ、マスターHBCモジュールだけが電源供給をイネーブルできる。モジュールが誤動作を始め、コマンドに対応しないと、HBCモジュールはそのモジュールへの電源供給をディスエーブルする。HBCモジュール用の制御第二レイヤは、カード管理バス(CMB)である。各モジュールはCMBに接続されるATMEL AT90S8515 (AVR) マイクロコントローラを有し、HBCモジュールそのものはマスターまたはスレーブとして動作するCMBに接続されるAVRマイクロコントローラを有する。CMBマイクロコントローラは、モジュール上のメインプロセッサに供給される電源とは別に、ミッドプレーンへの接続によって電源供給される。CMBにより、マスターHBCはカードタイプを読み出し、カードの有無を判断し、カードにマスク不可割込を送るか、あるいはカードのハードリセットを行うことができる。モジュールプロセッサとマスターHBCモジュールは、モジュール上のAVRマイクロコントローラのシリアルポートを通じた通信を行うこともできる。この通信パスは、PCIが故障した場合に通信を制御するためのバックアップとなる。

【００８１】HBCモジュール用の制御第三レイヤはPCIバスである。モジュールがPCIバス上の制御プロセスを使って反応しない場合、CMBを通じて質問することができる。それでもモジュールが反応しない場合は、CMBを使ってマスク不可割込を送信し、それでも反応がない場合は、CMBを通じてリセットする。リセット後も依然としてモジュールの反応がない場合、電源を落とし、モジュールを交換せよとの警告を発することができる。[HBCモジュールリダンダンシー]HBCモジュールリダンダンシーとフェイルオーバはシステムリダンダンシーをサポートする。HBCモジュール202A-Bを両方同時に動作させることは可能であるものの、HOST_SEL信号によってマスターに指定できるのは一方だけである。マスターHBCモ

ジュールはPCIバスの全部にPCIバスアービトレーションを供給し、他のモジュールへの電源イネーブルを制御し、CMBデバイス上のマスターとして認識される。バックアップHBCモジュールのPCIバスアービトレーション信号と電源イネーブルは、HOST_SEL信号によってディスエーブルされる。CMBはカードのスレーブCMBまたはFCBデバイスのそれぞれで、HOST_SEL信号によって切り換えられ、HOST_SEL信号は抵抗によってシステムミッドプレーン(SMP)上でプルダウンされ、HBCモジュール202Aがデフォルト時のマスターとなる。HBCモジュール202BはHOST_SEL信号によって自分をマスターとすることもできるが、このようになるのは普通、フェイルオーバ時あるいはHBCモジュール202Aがない場合の立ち上げ時だけである。

【００８２】エラー発生の可能性をなくすため、FVCはHOST_SEL信号を駆動し、特定パターンを2ヵ所の離れたメモリ位置へ書き込むことを要求する。これにより、誤動作しているHBCモジュール自身がマスターになることが防止される。HBCモジュールの電源イネーブル信号はどちらもSMP上で引き下げられ、立ち上げ時に両方のカードに電源が供給されるようになる。HBCモジュール202AはHBCモジュール202Bへの電源供給イネーブルを制御し、これと同様にHBCモジュール202BはHBCモジュール202Aへの電源供給イネーブルを制御する。再び、エラー発生の可能性をなくすため、HBCモジュールの電源供給イネーブル信号の駆動には、特定パターンを2ヵ所の離れたメモリ位置へ書き込むことが必要となる。PCIブリッジはデュアルホストをサポートしていない。PCIブリッジを特別に構成することにより、両方のHBCモジュールをシステムPCIバス上に構成することができる。両HBCモジュール上のPCIブリッジは、ひとつのHBCモジュールが制御するアドレススペースが他のHBCモジュールのPCIでリッジ上のシステムPCIバス全部にとってローカルなメモリスペースとしてマップされるように構成される。ひとつのHBCモジュールが他のPCIアドレススペースを読み出し、これに書き込もうとすると、エラーか発生する。また、システムPCIバスへの4つのブリッジが重大なエラーの原因となるトランザクションを認識するとエラーが発生する。したがって、ひとつのHBCモジュールは、システムバス上の他のHBCモジュールへのアクセスを試みるべきではない。

【００８３】HBCモジュールはPCIバス上で通信すべきではないが、HBCモジュールは2つの別個の通信用バス、専用シリアルポートというバスを有する。専用シリアルポートは通信用プライマリパスとなってメッセージを伝え、他のHBCモジュールのサニティチェックを行う。シリアルポートが故障した場合、CMBをバックアップとして使い、どのHBCモジュールが故障したかを判断することができる。［HBCモジュール立上げシーケンス］HBCモジュールはどちらも、システム電源投入時にEVCによってパワーアップされるため、パワーアップ時に他のHBCモジュールがあるかどうかを判断する必要があるが、これはCMBを通じて行われる。他にもモジュールがある場合、HBCモジュール202Aはデフォールト時にマスターとなる。パワーアップ時にHBCモジュール202AがHBCモジュール202Bはないと判断した場合、HBCモジュール202Bのカードスロットへの電源供給をディスエーブルすることができる。これにより、第二のHBCモジュールを追加し、マスターHBCモジュールのコントロール下でパワーアップされる。HBCモジュール202Aが、HBCモジュールが存在すると判断した場合、シリアルポートを通じた通信が行われる。パワーアップ時にHBCモジュール202BがHBCモジュール202Aはないと判断すると、202B自身がHOST_SEL信号をセットし、HBCモジュール202Aのカードスロットへの電源供給をディスエーブルすることにより、マスターHBCモジュールとなる。HBCモジュール202BがHBCモジュール202Aの存在を判断すると、HBC 0がシリアルポートを通じた通信を行うまで待たなければならない。所定の時間が経過しても通信が行われない場合、HBCモジュール202Bはフェイルオーバシーケンスを開始する。［HBCモジュールのフェイルオーバシーケンス］HBCモジュールはシリアルシンターフェース中、特定の間隔で相互に通信するはずである。バックアップHBCがマスターHBCとのシリアル通信を行わなくなると、そのCMB上でマスターHBCモジュールとの通信を確立するよう試みるべきである。CMB上で通信が確立され、両方のホストが健全であると、シリアル通信リンクが異常である。両方のカードは、どこに故障があるかを判断する診断を行わなければならず、故障がバックアップHBCモジュール上にある、または分離できない場合、アラームをトリガーする。故障がマスターHBCモジュール上にある、またはCMB通信が確立できない場合、バックアップHBCモジュールはマスターHBCモジュールの電源を切り、自分がマスターとなる。［ソフトウェアアーキテクチャの概要」ストレージサーバは、他にないような広帯域、高スループットおよびストレージサーバのデマンドをサポートするように設計されたオペレーティングステムによってサポートされる。オペレーティングシステムは、バスシステム上のデータ転送をスケジュール、制御し、システムを管理する。多数の異なるオペレーティングシステムとソフトウェアコンポーネント構成を利用できるものの、ある実施形態においては、ストレージサーバ用に設計されたモジュール性の高いオペレーティングシステムを使用する。

【００８４】図7は、ストレージサーバ用のオペレーティングシステムとサポーティングプログラムのソフトウェアモジュールを示すブロック図である。

【００８５】図7は、ハードウェアインターフェースモジュール900、アラバマ州モビールのAccelerated Technologies社製Nucleus Plus™リアルタイムカーネルモジ

ュール902、ISOSプロトコル管理モジュール904、ストレージサービスモジュール906というオペレーティングシステムコンポーネントを有する。ハードウェアインターフェースモジュール900により、ストレージサーバのソフトウェアコンポーネントはストレージサーバのハードウェアコンポーネントと通信することができる。

【００８６】Nucleus Plus™リアルタイムカーネルモジュール902は、タスク、キュー、信号、タイマー、重要セクションサポートといった基本的オペレーティングシステム機能を提供するのに使用され、ストレージサービスモジュール906によってC++クラスの機能としてストレージサーバのソフトウェアモジュールにエクスポートされる。

【００８７】ISOSモジュール904により、ストレージサーバは入出力用のメッセージングアーキテクチャをサポートする。RAIDコントローラ(RAC)モジュール、ネットワークインターフェースコントローラ(NIC)モジュール、ソリッドステートドライブ(SSD)モジュール、ディスクドライブハブ(DDH)モジュール、光ファイバチャネルハブ(FCH)モジュール等のハードウェアモジュールはすべて入力/出力プロセッサ(IOP)である。マスターホストブリッジプロセッサ(HBC)モジュールは、ホストとなる。

【００８８】ストレージサービスモジュール906は、メッセージングクラスを使ってコンポーネント間の信頼性の高いメッセージ転送をサポートし、デバイスドライバモジュールの動作と仮想デバイス用サポートをサポートする。デバイスドライバモジュール(DDM)と仮想デバイス(VD)はストレージサーバストレージシステムの構成ブロックである。ストレージサービスモジュール906は、ストレージトランザクションに対するリクエストをサポートするように構成されている。

【００８９】いくつかのアプリケーションにおいては、ストレージサーバ102A等、単独のストレージサーバがオペレーティングシステムモジュール900-906と一緒に作動する数百のDDMを有し、ストレージサーバリクエストに対応する。また別のアプリケーションでは、わずかなDDMをいろいろ組み合わせて使用する。

【００９０】ソフトウェアコンポーネントはデバイスドライバモジュール(DDM)として実装されている。主としてハードウェアデバイスにリクエストを送るDDMは中間ドライバモジュール(HDM)と呼ばれ、内部の中間プログラムとして機能するDDMは中間サービスモジュール(ISM)と称される。例えば、SSDモジュールに働くDDMはHDMと呼ばれ、キャッシュ、ミラーリングおよびその他ハードウェアデバイスに直接連結されていないサービスを提供するDDMはISMと呼ばれる。

【００９１】単独のDDMが単独のストレージサーバ上で複数の例示を有することもある。例えば、図７において、動作、健全性、ステータスPHSモニター908A-Dという4つの例示があり、それぞれNIC 910, RAC 920, HBC 930, SSD 940という4つの主要なソフトウェアサブシステムのいずれかに対応する。各DDMは専用のメッセージキューと個別の識別子を有する。例えば、NIC 910上のPHSモニター908Aは、デバイスID (DID) 0となる。各DDMは、DDMが扱うストレージリクエストのクラスをリストし、オペレーティングシステムモジュールは、ストレージリクエストのクラスに基いてDDMにリクエストをルーティングする。リクエストは、リクエストコードまたは仮想デバイス番号によってルーティングできる。

【００９２】NICソフトウェアサブシステム910は、プロセッササポートHDM 912A, 入力/出力変換ISM 914A およびPHSモニター908Aという3つのDDMを、RACソフトウェアサブシステム920は、プロセッササポートHDM 912B, 入力/出力変換ISM 914BおよびPHSモニター908Bという3つのDDMを、HBCソフトウェアサブシステム930はプロセッササポートHDM 913C, 入力/出力変換ISM 914C、カード管理HDM 916、システムモニターDDM 918、インターネットプロトコルDDM 921、フロントパネルディスプレイDDM 922、特定用途向けプロセッササポートDDM 924、PHSモニター908Cを有する。SSDソフトウェアサブシステム926は、ソリッドステートドライブ管理HDM926とPHSモニター908を有する。フロントパネルディスプレイ950はハイパーテキストマークアップ言語(HTML)クライアント928をサポートする。

【００９３】図８～１０はさまざまなハードウェアドライバモジュール(HDM)を示し、図１１～１４は本発明の好ましいアーキテクチャによる各種の内部中間サービスモジュール(ISM)を示す。図１５は仮想回路となるデータパスに構成されたドライバモジュールセットの簡略図である。

【００９４】図8は、HDM 524を有するネットワークインターフェースカード520を示す。カード520は光ファイバチャネルネットワークへの物理的インターフェース521を有する。この例においてはカリフォルニア州コスタメサのQlogic Corporation社製ISP 2200A等のQlogicデバイスであるネットワークインターフェースチップ522は、物理的インターフェース521に接続され、ライン523で表わされる通信を発生し、これがHDM 524の中で処理される。HDM 504はシステム内の他のドライバモジュールが使用するよう、通信の条件付けを行い、ライン525によって表わされる通信はSCSIフォーマットを有する。ライン526が示す通信は、BSAフォーマットなどのメッセージフォーマットを有し、ライン527が示す通信はインターネットプロトコル(IP)フォーマットを有する。HDMは図中、「Qlogicドライバ」と表示されたドライバクラスの例であり、この例ではデバイス識別子DID 401が与えられている。物理的インターフェースはNIC #1として識別される。

【００９５】図9は、不揮発性集積回路メモリデバイス

のアレイで実装されるストレージデバイス720を示す。HDM 722はアレイ721と接続され、ライン723でのブロックストレージアーキテクチャ通信をアレイ721からの記録再生用フォーマットに変換する。この例では、HDM 722にはデバイス識別子1130が与えられ、物理的インターフェースはSSD #4として識別される。

【００９６】図１０は、図６に示す好ましい実施形態における光ファイバチャネルアービトレーティドループアーキテクチャのストレージサーバシャーシに設置されたディスクドライブアレイ820の構成を示す。図６に示される光ファイバチャネルディスクハブ#0 216A、チャネルディスクハブ#1 216B、光ファイバチャネルディスクハブ#2 216C、光ファイバチャネルディスクハブ#3 216Dは、冗長ハブコントロールHDM 821，822に接続されている。

【００９７】HDM 821，822はそれぞれ物理的光ファイバチャネルアービトレーティドループ接続823，824に接続される。HDM 821にはデバイス識別子1612、HDM 822にはデバイス識別子1613が付与されている。接続823は光ファイバチャネルインターフェース825に接続され、インターフェース825は、物理的インターフェース840とHDM827に接続されるネットワークインターフェースチップ826を有する。ISM 828はHDM 827と内部通信バス829に接続される。ISM 808は、ライン829上のブロックストレージアーキテクチャ通信をHDM 827用のIOCB通信に変換する。HDM 827はネットワークインターフェースチップ826と通信し、チップ826は光ファイバチャネル823を駆動する。ISM 828にはデバイス識別子1210、HDM 827にはデバイス識別子1110が付与される。物理的インターフェース825はRAC #0とラベリングされる。

【００９８】光ファイバチャネル接続824は、インターフェース830に接続され、インターフェース830はインターフェース825と同様の構成であり、ネットワークインターフェースチップ832によって駆動される物理的光ファイバチャネルインターフェース831を有する。ネットワークインターフェース832は、ライン833で示すチャネル上でHDM 834と通信する。HDM 834はチャネル816を通じてISM 835と通信し、ISM 835はチャネル837上のBSAフォーマットメッセージへのインターフェースを管理する。この例において、ISM 835にはデバイス識別子1211、HDM 834にはデバイス識別子1111が付与され、インターフェース830はRAC #1として識別される。

【００９９】図１１～１４は、データパスに構成することのできる、本発明によるISM の例をいくつか紹介したものである。

【０１００】図１１は本発明によるプロトコルサーバモジュールの一例であるSCSIターゲットサーバ550を示す。本発明のストレージサーバを通じて管理されるデータのユーザが利用する特定のストレージチャネルまたはネットワークプロトコル用に、同様のプロトコルサーバモジュールを利用することができる。ターゲットサーバ550は、ユーザとの接続用の通信インターフェースに接続された、図８のHDM等、HDMから入ってくるメッセージを受け取るメッセージインターフェース551を有する。この例においては、インターフェース551上のメッセージはSCSIフォーマットを有し、別の例でメッセージはすでにBSAアーキテクチャあるいは現在使用中の通信インターフェース上のプロトコルに適した別のアーキテクチャを持っているかもしれない。サーバ550は、SCSI－BSAトランスレータ553あるいはアンサーローカル機能554に入るメッセージを変換するスイッチ機能550を有する。通常、トランスレータ553はメッセージをライン555上の外に出て行くメッセージとして送る。ライン555上の中に入るメッセージはトランスレータ556に供給され、これが入ってくるBSAメッセージをライン551で用いられるSCSIフォーマットに変換する。

【０１０１】多くの例において、SCSIターゲットデバイスは、さらにメッセージをルーティングすることなくローカルアンサーサービス554を使ってSCSIメッセージに応えることができる。ストレージそのものからの読み出しまたは書き込みに関係のない多くのステータスメッセージはローカルアンサーサービス554が扱う。

【０１０２】この例におけるターゲットサーバ550は、クラスSCSIターゲットサーバの例であり、デバイス識別子500が付与される。SCSIターゲットサーバ550といったプロトコルサーバの機能のひとつは、関連するインターフェース上でのストレージトランザクションの対象となるストレージ範囲を識別することである。ストレージ範囲は、以下に詳述するストレージサーバ内の構成ロジックを使って仮想回路にマップされる。

【０１０３】図１２は、ミラー管理データパスタスクを実行するISM 650を示す。ISM 650はデバイス上の内部通信チャネルに接続されるインターフェース652を有する。論理プロセッサ652は入ってくる通信およびデータを受け取り、ミラーリング機能を管理する。ロジック652はプライマリドライブ653、セカンダリドライブ654、第三のドライブ655および予備ドライブ656を含む複数のドライブインターフェースと通信する。図中では3方向ミラーリングが示されているが、仮想回路を使い、所望の数のミラーパスを作り、「N方向」ミラーリングを行うこともできる。「ドライブインターフェース」という言葉を用いているものの、ミラーリング機能には他のストレージデバイスも利用できる。ドライブインターフェース653-656は、内部通信チャネルを用いて、ミラーリング機能で使用されるターゲットストレージデバイスと関連するHDMモジュールと、あるいは特定の仮想回路に適したその他のISMモジュールと通信する。この例において、ミラーISM 650は「ミラー」というクラスの例として実装され、デバイス識別子10200を与えられている。

【０１０４】図１３はパーティションISM 750を示す。

パーティションISM 750は、他のドライバモジュールから内部通信を受信するインターフェース751および他のドライバモジュールとも通信するインターフェース752を有するほか、ロジックプロセス753、ベースアドレス754とリミットアドレス755を記憶するデータ構造、ドライブインターフェース756を備えている。パーティションロジックプロセス753は、各種ストレージ管理技術に役立つ論理パーティショニング機能を用い、ドライブプロセス756によって識別されるサブジェクトストレージデバイスを構成するため、物理的デバイスが仮想回路における複数の論理デバイスのように見える。この例において、パーティションISM 750は「パーティション」というクラスの一例であり、デバイス識別子10400が付与されている。

【０１０５】図１４はキャッシュISM 850を示す。キャッシュISM 850は、ストレージサーバ上の内部メッセージ転送構造へのインターフェース851と通信する論理プロセッサ853を有する。キャッシュISM 850におけるデータ構造には、ローカルキャッシュメモリの割り当て854、キャッシュ854に保存されたデータを識別するキャッシュテーブル855、ドライブインターフェース856を有する。ドライブインターフェースはチャネル857上で、キャッシュが使用中の特定の仮想回路に関連するHDMと通信する。一実施形態におけるキャッシュメモリ854はストレージサーバの中で管理され、別の実施形態において、キャッシュは図９について説明したようなアーキテクチャを有するソリッドステートメモリモジュール等の高速不揮発性メモリの中に保存することができる。好ましい実施形態において、キャッシュISM 850は「キャッシュ」というクラスの一例として実装され、デバイス識別子10300が付与される。

【０１０６】図１５は、本発明による複数のドライバモジュールを有する、データパスによって実装される冗長仮想回路の発見的図式である。仮想回路はデータのユーザと通信するための外部インターフェース、ユーザとの通信をドライバモジュールの通信フォーマットに変換するためのプロトコルトランスレータ、ストレージデバイスとの通信インターフェースを含むストレージオブジェクトを有する。データパスタスクを行うストレージオペレータは、トランスレータとストレージオブジェクトの間に設置できる。キャッシュ、ミラー、パーティション等のストレージオペレータとして動作するドライバモジュールの配列は、システムデザイナがストレージサーバによって提供される構成済みロジックを使って最適なものとする。

【０１０７】図１５に示す例において、外部インターフェースはNIC #0によって提供され、これに関連するHDMはブロック1010が示す。プロトコルトランスレータはSCSIターゲットサーバISM 1011、キャッシュ機能はISM 1012、ミラー機能はISM 1013である。ストレージオブジェクトはミラー機能1013からアクセスされ、この例においてはブロック1014で示す光ファイバチャネルの基本的デイジー・チェーンインターフェースとその関連HDMあるいは外部LUNインターフェースから選択された物理的ストレージインターフェースセット、ブロック1015および冗長ブロック1016が示すISN/HDMのペアを通じてアクセスされる光ファイバチャネルアービトレーティドループの中のディスクドライブ、ブロック1017が示すソリッドステートストレージデバイスと関連HDM、ブロック1018が示す外部ディスクドライブとのインターフェースおよびこれに関連するISM/HDMのペアで構成される。ディスク(01)，(02)，(03)，(04) 上の個別のHDMモジュールは光ファイバチャネルアービトレーティドループを通じたインターフェース1015および1016との通信を管理する。

【０１０８】この実施形態において、ミラーモジュール1013はディスク(01)，(02)，(03)にそれぞれプライマリ、セカンダリ、予備ドライブとしてアクセスし、ミラー機能を果たす。図１２に示すミラーモジュールには第三のドライブインターフェースが含まれているが、図１５のシステムはこの第三のドライブを使用していない。

【０１０９】この図には、ISMモジュール1020と1021も描かれており、これらは図中の仮想回路のデータパスとは接続されていない。これらのブロックは、仮想回路構造を使用すると、単純にストレージサーバを構成することにより、パーティショニング等の新しいモジュールを追加することができることを示すものである。

【０１１０】冗長データパスはブロック1025で示すインターフェースNIC #1とこれに関連するHDM、ブロック1026で示すSCSIターゲットサーバISM、ブロック1027で示すキャッシュISM、ブロック1028で示すミラーISMで構成される。データストレージデバイスのリダンダンシーは、ミラー機能を使って実現している。冗長ドライバモジュールは、好ましい実施形態において、ストレージサーバにおける個別のIOPの上に分散されている。

【０１１１】図１５に示すように、ドライバモジュールはそれぞれ、図１５中のブロック内で括弧書きされた個別のドライバ識別子を有し、ストレージサーバが管理し、ストレージサーバ内の構成可能なロジックで制御するコンフィギュレーションデータベース内のテーブルに基いて構成ロジックをサポートするのに、このデバイス識別子が用いられる。

【０１１２】好ましいシステムにおいて、構成テーブルは図１６および１７に描かれているような持続型テーブルドライバによって管理される。図４に戻ると、ストレージサーバ102はテーブル116のようなテーブルにおける管理およびルーティング情報を記憶する。テーブル116は管理インターフェース120からアクセスできる。テーブル116は通常、不揮発性メモリ等の持続型メモリの中に記憶され、冗長的に保持されてフェイルセーフをサポートする。

【０１１３】図１６は、ドライバモジュール構成の基本的アーキテクチャによる、「持続型テーブル」というクラスの一例として実装された持続型テーブルモジュール1400を示す。この持続型テーブルモジュール1400は、テーブルアクセス論理プロセッサ1401、テーブルデータアクセスマネージャ1402を含む各種サポート機能、持続型イメージマネージャ1403、および持続型テーブルインスタンス同期モジュール1404からなる。テーブルデータアクセスマネージャ1402はこの実施形態において、テーブルクラスマネージャ1405と接続され、このテーブルクラスマネージャは、光ファイバチャネルポートIDテーブル1406、LUNエクスポートテーブル1407、構成テンプレートテーブル1408、DDMロールコールテーブル1409、仮想デバイステーブル1410、ストレージロールコールテーブル1411、光ファイバチャネルディスクロールコールテーブル1412、外部LUNテーブル1413、ソリッドステートストレージテーブル1414を含む複数の構成テーブルを管理する。持続型テーブルモジュール1400が管理するテーブルセットの特定の構成は、特定の実装に合わせて変更し、あるクラスのデバイスにとって最適なものとすることができる。

【０１１４】持続型イメージマネージャ1403とテーブルインスタンス同期マネージャ1404は、図１１に示すような持続型データストレージドライバ1420および図示されていない第二の持続型ストレージドライバと通信する。持続型データストレージドライバ1420はHDMとして実装され、これは「持続型ストレージ」というクラスの一例であり、先に説明したドライバモジュールのモデルに従ってデバイス識別子が付与されている。好ましいシステムにおいて、持続型データストレージHDM 1420は、ストレージサーバ内のソリッドステートストレージデバイスと通信し、仮想回路で用いられるデータに高速アクセスすることができる。

【０１１５】持続型データストレージはこのシステムに関するさまざまな構成情報を保持する。DDMロールコールテーブル1409には、すべてのデバイスドライバモジュール例リストとそれぞれの固有のデバイスIDが含まれる。ストレージロールコールテーブル1411には、ストレージサーバが検出する全能動ストレージデバイスリストが含まれ、このロールコールテーブルは仮想デバイステーブル1410と構成ツールが仮想回路を作成するのに使用する。LUNエクスポートテーブル1407は、ストレージチャネルトランザクション内の特定されたストレージ範囲を仮想回路にマップできるようにする。外部LUNテーブル1413は、ストレージサーバ上の外部ストレージインターフェースを通じて接続されるその他のストレージサーバの中に保持されるストレージの論理ユニットを識別する。

【０１１６】２つのプライマリテーブルがクライアントへのストレージのエクスポートと、ストレージサーバ102Aのストレージルーティング機能をサポートする。これらのテーブルは、エスクポートテーブル1407と仮想デバイス構成テーブル1410である。

［エクスポートテーブル1407］エクスポートテーブル1407は、ストレージトランザクションとともに受け取ったアドレシングジ用法を仮想回路またはストレージオプションにマップする。光ファイバチャネルインターフェース上のSCSI-3の場合に使用されるアドレシング情報は、イニシエータID、ターゲットLUN、ターゲットアドレスである。

【０１１７】すべてのイニシエータ、あるいはクライアントが多くのLUNを共有するため、ひとつひとつのリクエストを解決するのに必ずしもこの情報のすべてを使用する必要はなく、ほとんどのLUNは、異なる仮想回路を選択するためよりもむしろ仮想回路内のアドレシングを行うために、ターゲットアドレス；例えばストレージデバイス上のオフセットを使用する。この代表的実施形態において、エクスポートテーブル1407は表１のように構成される。

【表１】

プロトコル	プロトコル別アドレッシング(LUN)	イニシエータ別？Yesであればid	回路内で最初の仮想デバイス	プライマリコネクションのオーナ
SCSI	0	No	11	NIC 0
SCSI	1	Yes,ID=6	30	NIC 0
SCSI	1	Yes,ID=5	60	NIC 1
SCSI	2	No	12	NIC 0
TCP/IP	port2000	No	70	NIC 0

【０１１８】エクスポートテーブル1407には、仮想回路の現状、仮想回路の容量その他の情報を記載する別のコラムを含めることができる。一実施形態において、エクスポートテーブル1407は、エクスポートテーブルのコラムで仮想回路全体を列挙する。

【０１１９】表１は、プロトコルごとのアドレシング情報を使ってリクエストを適当な仮想回路へルーティングできることを示している。従って、ポート2000をターゲットとするストレージ範囲を識別するものとして使用するTCPセッションだけが、識別子70を有する仮想デバイスから始まる仮想回路にルーティングされる。

【０１２０】表１は、あるプロトコルについてひとつのLUNを、ストレージトランザクションのイニシエータに応じて異なるデバイスに接続できることを示している。この例において、LUN 1はイニシエータIDに基いて異なる仮想回路にマップされる。また、仮想回路は、「ワー

ルドワイドネーム(WWN)」等、他の種類の識別子に基いてマッピングすることもできる。

【０１２１】エクスポートテーブルの一例を以下に示す:

```
#define EXPORT_TABLE  "Export_Table"
struct ExportTable Entry {
rowID  ridThisRow;        //表のこの行のrowID
U32    version;          //エクスポートテーブルバージョンの記録
U32    size;             //エクスポートテーブルサイズの記録(バイト)
CTProtocalType    ProtocolType:    //FCP, IPその他
U32    CircuitNumber;     //LUNその他
VDN    vdNext;           //パス内で最初の仮想デバイス番号
VDN    vdLegacyBsa;      //レガシーBSAの仮想デバイス番号
VDN    vdLegacyScsi;     //レガシーSCSIの仮想デバイス番号
U32    ExportedLUN;      //エクスポートされたLUN番号
U32    InitiatorId;         //ホストID
U32    TargetId;         //われわれのID
U32    FCInstance;       //FCループ番号
String32    SerialNumber;  //シリアル番号のストリングアレイを使用
Long long   Capacity;     //この仮想回路の容量
U32        FailState;
U32        PrimaryFCTargetOwner;
U32        SecondaryFCTargetOwner;
CTReadyState  ReadyState;  //カレント状態
CTReadyState  DesiredReadyState;  //所望の準備状態
String16    WWNName;    //ワールドワイドネーム(64または128ビットでIEEEに
登録されたもの)
string32        NAME;             //仮想回路名
}
#endif
```

[仮想デバイス構成テーブル]仮想デバイス構成テーブルは、仮想デバイスを、仮想デバイスをサポートするデバイスドライバに接続する。仮想デバイスは、冗長デザインをサポートするように設計されているため、仮想デバイス構成のためのテーブルは仮想デバイス番号をデバイスモジュールにマップする。一実施形態において、表2のようなテーブルを使って仮想デバイスがこれをサポートするデバイスドライバにマップされる。図１５は、表２で実装される仮想デバイス12から始まる仮想回路を示したものである。

【表２】

仮想デバイス	プライマリ	代替	パラメータ	ステータス	クラス
1	4000	4001	N/A	プライマリ	持続テーブル
10	1210	1211	SO(00)	代替	FC ディスク
11	500	501	VD(10)	プライマリ	SCSIターゲット
12	500	501	VD(13)	プライマリ	SCSIターゲット
13	10300	10301	VD(14)	プライマリ	キャッシュ
14	10200	10201	VD(15,16,null,17)	プライマリ	ミラー
15	1210	1211	SO(02)	プライマリ	FC ディスク
16	1210	1211	SO(03)	プライマリ	FC ディスク
17	1210	1211	SO(04)	プライマリ	FC ディスク

【０１２２】表2のように、各仮想デバイスについて、その仮想デバイスをサポートするプライマリおよび代替ドライバモジュールに関する情報が提供される。例えば、表2の2行目では、光ファイバチャネルディスクドライブが仮想デバイス(VD)10にマップされる。

【０１２３】仮想デバイスは仮想デバイスをサポートする一つ以上のソフトウェアあるいはハードウェアモジュールで構成される。パラメータのコラムは、初期化情報を提供するのに使用される。VD 10の場合、パラメータはSO(00)で、これはストレージオプション0を意味する。各デバイスドライバモジュールのクラスは、クラスごとのパラメータを有する。ストレージオプションドラ

イバは、特定のストレージユニットを指定するパラメータを使い、ミラードライバやキャッシュドライバといった中間ドライバクラスは、仮想回路内の次の仮想デバイスを指定するパラメータを使用する。このフォーマットによれば、ひとつのデバイスドライバモジュールがパラメータの設定に基いて複数のデバイスをサポートすることができる。表２においては、デバイスドライバ1210は仮想デバイス10，15，16，17によって使用されているが、それぞれドライバに異なるパラメータを指定する。ステータスのコラムは、仮想デバイスをサポートするソフトウェアまたはハードウェアモジュールのステータスを示す。例えば、表２の１行目では、ステータスは「プライマリ」であり、これはプライマリデバイスドライバ、つまりここでは4000が使用されることを意味している。表２の２行目のステータスは「代替」であり、これはプライマリデバイスドライバが故障した、あるいは正しく応答していないことを示す。この場合、代替ドライバ、つまり表２の２行目では1211が使用される。複数の代替を有するデバイスの場合、ステータスのコラムには使用されているドライバが表示される。

【０１２４】例）例えば、接続オプション130のいずれかひとつの上で、SCSIプロトコルを使い、アドレシング情報の中でLUN 2を指定して、ストレージサーバ102Aへとストレージトランザクションが行われる場合について考えてみる。この例において、ストレージサーバ102Aは表1および2のように構成されているものとする。

【０１２５】ストレージトランザクションを受け取るネットワークインターフェース146等の接続オプションは、ハードウェアデバイスドライバに接続される。ハードウェアデバイスドライバはストレージトランザクションを受け取り、プロトコルに従って、そのプロトコルを扱う適当な仮想デバイスにこれを送る。

【０１２６】例えば、SCSIストレージトランザクションは、SCSIターゲットクラスの中のデバイスドライバに送られる。同様に、IPストレージトランザクションは、IPターゲットクラスの中のデバイスドライバに送られる。ここで、ストレージトランザクションはSCSI通信プロトコルを使って作られたため、SCSIターゲットデバイスドライバ(DID 500)にルーティングされる。

【０１２７】SCSIターゲットデバイスドライバはさらにリクエストを分析する。分析ではまず、そのリクエストをどの仮想回路にマップするかを判断する。この判断は、エクスポートテーブル内の情報を使って行われる。この例において、表１は、LUN2を指定するSCSIプロトコルを使用するリクエストは仮想デバイス12から始まる仮想回路にルーティングされるべきであることを示している。一実施形態において、SCSIターゲットリクエストはすべて、ひとつのインターフェースに関する同じSCSIターゲットドライバにルーティングされ、この実施形態では、ターゲットVD 12のパラメータ情報は、SCSIターゲットの第二の仮想デバイスにメッセージをルーティングするよりも、SCSIターゲットデバイスの行動を制御するのに使用される。

【０１２８】ここでドライバ番号500とされているSCSIターゲットデバイスはSCSIメッセージを内部フォーマットに変換する。このようなフォーマットの一例が、I_2O ブロックストレージアーキテクチャ(BSA)フォーマットに基くものである。このフォーマットはデバイスおよびプロトコルニュートラルであり、中間デバイスドライバはこれを使用できる。リクエストが内部フォーマットとなると、これはパラメータによって示される仮想回路内の次の仮想デバイスに送られる。ここでは、このパラメータはVD(13)、つまり仮想デバイス13である。

【０１２９】メッセージはVD 13にルーティングされ、これは冗長キャッシングドライバであり、ここでは10300および10301とナンバリングされている。キャッシングドライバはメモリを使ってストレージトランザクションを記憶する。ドライバが使用しているキャッシングアルゴリズムに基いて、ドライバは適当な間隔でストレージトランザクションを仮想回路内の次の仮想デバイスにルーティングする。ここで、次のデバイスはパラメータVD(14)、つまり仮想デバイス14によって示される。

【０１３０】内部フォーマットにおいて、メッセージはVD 14にルーティングされる。仮想デバイス14は冗長ミラーリングドライバを有する。この場合、ドライバ10200および10201が使用される。ミラーリングドライバは、複数のボリュームでミラーリングされたストレージイメージを保持するためのミラーリングアルゴリズムを実現する。このミラーリングドライバは、プライマリ、セカンダリおよび第三のストアおよび予備ストアをサポートしているが、他のミラーリングドライバは異なるアルゴリズムをサポートすることもある。このミラーリングドライバは、既存のストアと確実に同期される新規ストアを接続する場合もサポートしている。ドライバが使用しているミラーリングアルゴリズムとミラーリングされたストアのステータスに基き、ドライバはストレージトランザクションを仮想回路内の適当な仮想デバイスにルーティングする。プライマリおよび代替ストアがどちらも機能しているのであれば、ミラードライバはパラメータVD (15，16，なし、17)または仮想デバイス15，16によってのみ、このリクエストをプライマリおよびセカンダリストアにルーティングする。パラメータリストの中の「なし」とは、この仮想デバイスについては現在、第三のドライブが使用されていないことを示す。

【０１３１】ミラーリングドライバは、2つのデバイスに逐次的あるいは並列にストレージトランザクションメッセージをルーティングすることができる。この例において、仮想デバイス15へのメッセージ送信が検討されるが、セカンダリストア、仮想デバイス16にまでこの例を拡張することが可能である。仮想デバイス15は、光ファ

イバチャネルドライブを制御するための冗長ドライバを有する。ドライバは、例えばBSAからSCSIへ等、内部フォーマットをドライバが使用するフォーマットに変換する。ドライバはまた、ドライブにアドレシング情報も提供する。ここで、パラメータSO(02)を使って、ストレージオプション、この例では光ファイバチャネルドライブ番号2が選択される。

【０１３２】このように、ストレージプラットフォームにおいて、ハードウェア機能(ディスクまたはフラッシュメモリ等)とソフトウェア機能(RAIDストライプまたはミラー等)はすべて、ほとんどの場合にデバイスと呼ばれるソフトウェアドライバを通じてアクセスされる。

【０１３３】これらのデバイスはペアにされ(このペアを構成する各々のデバイスは別個の基板上で動作し、冗長性を持たせることが好ましい)、仮想デバイスと呼ばれる。次に仮想デバイスが連結され、各種の構成が出来上がる。例えば、ミラーデバイスは2つまたは3つのディスクデバイスに連結できる。このような構成を通じて、仮想デバイス連鎖が完成する。これらの仮想デバイス連鎖は、別の構成でも使用可能なBSAタイプのデバイスに構成されている限り、追加できる。

【０１３４】仮想デバイス連鎖はFCP/SCSIターゲットサーバデバイスに接続され、FCPターゲットドライバの「エクスポート」に関するLUNエクスポートテーブルの中にマッピングされる(つまり、外部からはFCPプロトコルを通じてアクセスされる)。この時点で、先頭にSCSIターゲットサーバデバイスを有する仮想デバイス連鎖は、仮想回路と呼ばれる。

【０１３５】仮想回路の構成を司る仮想回路マネージャソフトウェアは、SCSIターゲットサーバの「先頭」を仮想デバイス連鎖に置き、その後FCPターゲットのエクスポートテーブルを更新することによって仮想回路をエクスポートする。このソフトウェアはまた、削除、休止、フェイルオーバ動作もサポートする。

【０１３６】仮想回路マネージャソフトウェアは、各仮想回路内の全仮想デバイスを1ヵ所にまとめて記載する仮想回路テーブルVCTを保持する役割も担う。この情報は、フェイルオーバ、ホットスワップ、シャットダウン等、多くのシステム動作を実行するのに必要となる。

【０１３７】初期化を行う場合、仮想回路マネージャソフトウェアはVCTそのものを持続型テーブルストアの中で定義し、さらにVCTの挿入、削除その他の変更を聞き取る。

【０１３８】新しい仮想回路を作るためには、SCSIターゲットサーバを例示し、新しいLUNをマップ、エクスポートするのに必要な情報をVCT内の記録に入れなければならない。仮想回路マネージャは、VCTへの挿入を聞き取り、その回答を受け取ると、次の動作を行う:

1.新たに挿入された記録の情報を有効化しようとする。記録に無効な情報が含まれていると、そのステータスフィールドにはエラーが表示され、それ以上の動作は行われない。

2.新しく挿入された記録によって指定された仮想回路のLUNについて、新しいSCSIターゲットサーバデバイスを作る。

3.新たな記録のステータスを「例示」とする。

4.仮想回路に割り当てられるストレージは、ストレージロールコールテーブルで使用されるようにフラグが立てられる。

5.エクスポートテーブルが更新され、LUNが新しいSCSIターゲットサーバに送られる。

仮想回路内の記録が削除されると、仮想回路マネージャは以下の動作を行う:

1.まだ済んでいない場合は仮想回路を休止させ、休止と表示する。

2.仮想回路の発送データをエクスポートテーブルから取り除く。

3.仮想回路の記録から参照されるロールコール記録に不使用と表示する。

4.仮想回路に関連するSCSIターゲットサーバの例示を除く。

仮想回路マネージャは、VCTにおける「エクスポート」フィールドの変更を聞き取り、VCTの中のいずれかの記録における「エクスポート」フィールドが「正しい」と設定されると、仮想回路マネージャは以下の動作を行う:

1.FCPターゲットのエクスポートテーブルに必要な変更を行うことにより、仮想回路をエクスポートする。

2.エクスポート動作中に何らかのエラーに遭遇した場合、VC記録のステータスフィールドが設定され、「エクスポート」フィールドは正しい状態のままとなる。仮想回路がエクスポートされないと、「エクスポートされた」というフラグが「間違い」にセットされる。

【０１３９】仮想回路マネージャは、仮想回路テーブルの「休止」フィールドへの変更を聞き取る。VCTのいずれかの記録における「休止」フィールドが「正しい」にセットされると、仮想回路マネージャは次の動作を行う:

1.VCが現在エクスポートされている場合、そのエクスポートが停止し、「エクスポートされた」というフラグが「間違い」にセットされる。

2.仮想回路内の仮想デバイスのすべてに休止メッセージが送られる。

3.休止動作中に何らかのエラーに遭遇した場合、VC記録のステータスフィールドが設定され、「休止」フィールドは正しい状態のままとなる。つまり、仮想回路が休止されていないと、「休止された」というフラグが「間違い」にセットされる。

【０１４０】[ユーザインターフェース]ユーザインターフェースは、本発明によるストレージサーバを構成する際に表示、使用するためのデータ処理構造によって作ることができる。画像には、ロゴを表示するためのフィ

ールド、サーバのシャーシに関する基本情報を表示するフィールドおよびアイコンセットを有するウィンドウがあり、これらのアイコンを選択すると、管理アプリケーションを起動することができる。ハードウェアとソフトウェアを管理するルーチン、ユーザアクセスを管理するルーチン、そしてサーバ内の長いプロセスをモニターするルーチンはボタンによって開始される。本発明によれば、サーバに付けられるホストを定義する機能、エクスポートされたLUNを管理されたリソースにマップする機能、管理されたストレージを構成する機能もボタン操作で起動できる。

【０１４１】このウィンドウには、ユーザ名入力用フィールドとパスワード入力用フィールドを含むユーザログオンダイアログボックスも含まれている。

【０１４２】［ホットマネージャ］ユーザはボタン操作でホストマネージャを起動する。ここでは、ストレージサーバ用のホスト(サーバ)を決定するための、Javaベースのユーザインターフェース(UI)について説明する。管理ソフトウェアがウィンドウを開くと、ここには、構成、使用する上で利用できる各ホストに関するいくつかのコラムにホスト名、ポート番号、イニシエータIDおよび説明を、入力する表が表示される。これ以外のフィールドには、別のコラムに記載したネットワークインターフェースカード識別子および個別のホスト識別子が含まれる。好ましい例における個別のホスト識別子は、光ファイバチャネルホストに関するワールドワイド番号である。

【０１４３】ホストマネージャはストレージサーバのJavaベース管理アプリケーションのサブコンポーネントであり、これによってユーザはNICポートとイニシエータIDに名称と内容説明を割り当て、LUNの定義プロセスを進めることができる。一般的な機能はマウスのポップアップ、ツールバーボタン、アクションメニューを通じて利用し、例えば新規ホスト追加ボタン、ホスト変更ボタン、ホスト削除ボタン等を使って、既存のホストにアクセスしたり、新ホストを定義することができる。

【０１４４】ユーザインタフェースは、ホスト情報を表示するためのメニューとテーブル、あるいはその他のグラフィクスで構成される。ユーザがホストマネージャパネルに入ると、テーブルには既存のホストすべてが記入される。ユーザは、テーブル内の行を選択できる。各行には、１つのホストに関する情報が含まれる。次にユーザはホストの変更または削除を選択し、変更を選択すると、ダイアログボックスが現れ、ユーザはホスト名や内容を変えることができる。変更後、OKまたはキャンセルボタンを押す。OKを押すと、その変更がテーブル内に表示され、サーバに送られる。削除を選択すると、ダイアログボックスが現れ、削除されるホストを示すラベルとOKおよびキャンセルボタンが表示される。OKを押すと、そのホストの行がテーブルから削除され、サーバでもこの削除が行われる。追加を選択すると、ダイアログボックスが現れ、ユーザはホストに関するすべての情報を追加することができる。OKを押すと、その新ホストに関する新しい行がテーブルに追加され、サーバでもこの追加が実行される。コラムラベルをクリックすれば、コラムをソートできる。

【０１４５】［ストレージのマッピング］ユーザはストレージ管理ルーチンを起動することができ、このルーチンで表示される画像には、ストレージの要素を表示する階層化ツリーによる表示構成を示すウィンドウが含まれる。

【０１４６】ストレージの要素はこのツリー構造を使って定義される(例えば、ミラー→ストライプ→ディスク)。これにより、ユーザはストレージに関するそのユーザの考え方に合った、体系化された方法でそのストレージを構成することができる。ストレージ要素の代表的な種類を以下に示す:

- ミラー
- ストライプ
- 外部LUN
- 内部ディスク
- SSD
- ストレージコレクション
- ストレージパーティション

これらの要素をツリー状に組み立てることにより(例えばMicrosoft Explorerのようなツリーディスプレイを使用する)、ユーザは仮想回路で使用するストレージを前もって構成することができる。各要素はパーティションに分割でき、これらのパーティションを異なる方法で使用することが可能である。例えば、ストライプセットをパーティションに分割し、ひとつのパーティションをひとつのLUNとしてエクスポートし、別のパーティションをミラー内の１つのメンバーとして使用できる(これをさらに分割することも可能)。

【０１４７】ストレージ要素をパーティションに分割した場合、これらストレージコレクションの中に保存され、このストレージコレクションはパーティションに分けられた要素の子供となる。分割されていない要素については、このパーティションコレクションは存在しない。各パーティションは、それが分割しているストレージの種類、つまりミラーパーティションか、ディスクパーティションか等によって識別される。あるストレージ要素のパーティションは、その要素のパーティションすべてが利用できる(つまり、ストレージの全要素が不使用の状態)場合を除き、ひとつのパーティションにまとめることはできない。このために、ユーザはパーティションに分割されたストレージ要素のうち、使用されていないパーティションだけを有するものを選択し、「アンパーティション」ボタンを押す。

【０１４８】専用スペアがあると、専用スペアもストレ

ージコレクションの中に保存され、このストレージコレクションはこれらのスペアが割り当てられた要素の子供となる。

【0149】したがって、ストレージ要素はそれぞれが、子供としてパーティションコレクション、スペアコレクションおよび親となる要素を構成する実際のストレージ要素を持ちうる。

【0150】ストレージマネージャはある意味で、サーバ上の接続されたすべてのストレージをリストアップするストレージロールコールテーブルの内容を表わすものと言うことができる。利用可能な各ストレージ要素はストレージツリーの最上部として見られ、例えば、ミラーは利用可能として表示されるが、ミラーの枝を構成するストライプとディスクはそのミラーに属するため、利用できない。これらを別の場所で再び利用するには、そのミラーから(したがって、そのミラーから下のストレージツリーから)取り除く必要がある。一実施形態において、これはWindows NTファイルエクスプローラプログラムにおいてファイルをひとつのディレクトリから別のディレクトリに移動する場合と同様に、ドラッグ・アンド・ドロップによって行う。すべてのストレージ(使用、不使用)のツリーは、この例のディスプレイでは左半分に表示され、ストレージの各要素はその種類を示すアイコンを持っていたり、また名称やIDを特定する。

【0151】ツリーの下、ウィンドウの右側あるいはその他一般的な場所に、利用可能な(不使用の)ストレージが列挙される。これは、別のストレージ要素あるいは仮想回路が使っていないすべてのストレージのリストである。明白に使用されていないストレージの多くは一般スペアプールの中に設置されると予想され、この利用可能な(使用されていない)ストレージのリストは、ユーザが新しいストレージツリーを構成する時の材料となる不使用のストレージ要素を簡単に見つけることができるようにするための便宜上使用されると予想される。例えば、ソリードステートストレージデバイス(SSD)のパーティションはストライプセット(RAID 0)によってミラーリングされ、このパーティションとストライプセットはどちらも、これらがミラーリングされるまで、利用可能なリストの中に含められる。2つのメンバーからミラーが作られると、このミラーは仮想回路に組み込まれるまで、利用可能なリストの中に表示される。

【0152】右側には、ユーザがこれをマウスでクリックすることによって選択したツリーの要素に関する情報とパラメータが表示される。利用可能なリストに表示されたストレージ要素が選択されると、利用可能リストとストレージツリーの両方で選択される。

【0153】追加、削除機能が搭載されているため、エントリを作ったり、削ったりすることができ、さらに変更機能により、ユーザインターフェースで提供されるツールを使い、ユーザはツリーの中のストレージ要素に関する「所有者」、「最終修理日」、「内容説明」等のフィールドを変更することができる。ユーザが自分の追加しようとしているもの(ミラー、ストライプ、ディスク等)を特定すると、これに適当なコントロールセットが付与される。

【0154】内部ディスクと外部LUNについて、ユーザは名称、サイズ、あるいはメーカー等の項目を指定できる。ディスクは1個のハードウェアであり、自動的に検出されるため、内部ディスクを指定することは特殊のケースのように思われる。ユーザがディスクを追加するのは、後に追加する何らかのハードウェアのために「プレースホルダ」を入れておく場合に限られる。これはSSD基板についても行われる。

【0155】RAIDアレイの場合はどうなるかというと、ユーザは特定のRAIDレベルの(当初はミラーかストライプ)アレイを作りたいと指定し、そのアレイのメンバーとなるストレージ要素を指定することができる。これは、利用可能なストレージ要素のリストからエントリを選択することによって行われ、アレイ容量はそのメンバーの容量によって決定する。すると、アレイのメンバーとして使用されるストレージ要素には利用不可とのタグが付けられ(これらはアレイの一部であるため)、アレイそのものが利用可能なストレージのリストに追加される。各RAIDアレイには、メンバーのひとつが故障した場合のためにそのアレイに割り当てられる専用スペアを設けることもできる。

【0156】ストレージ要素のパーティショニングも可能であり、これはパーティションに分割すべき要素を選択し、ユーザがどのサイズのチャンクを希望するかを指定することによって行われる。その要素が過去に分割されていなければ、これによって2つのパーティションが作られる。つまり、ユーザが希望したパーティションとストレージの残り(不使用)のパーティションである。不使用部分からさらに別のパーティションも作られる。

【0157】各ストレージ要素に関する詳細な表示により、利用できる最大限の情報が得られる。好ましいシステムにおいて表示される項目のひとつは、特定のストレージ要素のパーティションがどのような種類のものか(大きさと位置)である。

【0158】[LUNのマッピング]ユーザインターフェースの1つのボタンを操作することにより、LUNマップルーチンが起動される。LUN(論理ユニット番号)マップは本質的にLUNとこれに関するデータのリストであり、名称と説明のリストとして表示され、そのLUNに関連するVC(仮想回路)がこのディスプレイ上に示される。ユーザがLUNマップからエントリをひとつ選択し、その詳細を求めると、これを見ることができる。LUNマップは、既存のLUNリストを、名称、説明その他のフィールドで表示する。これらのフィールドには以下のものがある:

- 名称

- 内容説明
- エクスポートされた状態
- ホスト
- ストレージ要素

LUNマップにより以下のことが可能となる：

- 各種フィールドに基くソーティング
- フィールドに基くフィルタリング。これは、一度に複数のLUNが動作する場合(例えば、イネーブル/ディスエーブル)のみ必要。
- 削除または編集/ビューのためにLUNを選択
- 新しいLUNの定義と追加
- 既存のLUNのインポート(ハードウェア立上げ時の「学習モード」で行われる)
- メンバーの追加とLUN上でのホットコピーミラープロセス開始
- LUNのエクスポートとアンエクスポート。これが基本的にホストからのデータの流れを開始、停止する。

仮想回路は、ボタン操作で起動できるストレージツリーあるいはホストに接続されるダイアログボックス等その他のグラフィック構成として(ユーザに対して)定義される。ダイアログボックスは、LUNの名称を入力するフィールド、内容説明を入力するフィールド、ターゲットIDを入力するフィールド、エクスポートされたLUNにンする情報を入力するフィールドを含む。ポップアップメニューは、利用可能なホストのリストの場合はホストボタン、利用可能なストレージ要素のリストの場合はストレージボタンで開くことができる。キャッシュ選択ボタンは、チェックボックスとして実装される。

【０１５９】ストレージツリーは実際にはストレージメンバーのツリーである(例えば、いくつかのストライプセットからなるミラーで、ストライプセットはいくつかのディスクからなる)。ホストは実際には特定のイニシエータIDを有し、NIC上の特定のポートに接続されるサーバである。ユーザはこれを、所定のホストおよび適量の利用可能なストレージを表わす所定のストレージツリーを選択することによって定義できる。

【０１６０】キャッシュの使用は、チェックボックスを使った「オン」または「オフ」に限定される。ベタのシステムでは、キャッシュのサイズやアルゴリズムの仕様に関するツールを提供する。キャッシュの使用は仮想回路に沿ったデータの流れを妨害することなく、実行中にオン、オフできる。LUNが作られた時のデフォルトは「オン」となる。

【０１６１】LUNマップの一実施形態では、仮想回路を作るのに必要な機能を有し、これはホスト用、ストレージ用2つのコラムを有するマルチコラムテーブルからなる。LUNを作るとこれが自動的にエクスポートされ、「追加」、「変更」、「削除」等の機能が利用できる。

【０１６２】LUNマップディスプレイで、ホットコピーミラーが定義される。これは通常、既存のLUNについて行われるからである。これは、LUNを選択してからミラーの追加を通じて既存のストレージツリーに追加するストレージツリーを選択すること、あるいは既存のミラーを拡張する(例えば2方向から3方向へ)のいずれかのプロセスとなる。

【０１６３】［データ移動のサポート］図１８は、通信リンク14では第一のストレージデバイス11に、また通信リンク14では第二のストレージデバイス12に接続されるストレージネットワークにおける3段階のデータ流れを示す簡略図である。中間デバイス10もまた、通信リンク13を通じてクライアントプロセッサに接続され、これによって中間デバイス10は論理アドレスLUN Aのデータにアクセスするためのリクエストを受け取る。

【０１６４】ストレージサーバ10はバッファとして使用される不揮発性キャッシュメモリ等のメモリ、リンク13上で受け取ったデータアクセスリクエストをリンク14と15でアクセスできるストレージデバイスに転送するためのデータ転送リソースのほか、本発明によるホットコピープロセスを管理するロジックエンジンを有する。このプロセスは、図１８に示す3つの段階を考えることによって理解できる。

【０１６５】ステージ1において、ストレージサーバ10は転送されるデータセットを特定し、リンク13で受け取ったすべてのデータアクセスリクエストをリンク14にマップしてデバイス11に接続し、このデバイス11がリクエストの対象となったデータセットを記憶する。ストレージサーバはホットコピープロセスを開始し、ターゲットデバイス、つまりこの例ではデバイス12を特定する制御信号を受信する。このステップによってステージ2が始まり、ステージ2でデータセットがバックグラウンドプロセスとして第一のデバイス11からストレージサーバ10を通じて第二のデバイス12に転送される。パラメータがストレージサーバ10上に保持され、このデータセットが転送される様子と、クライアントプロセッサからのデータアクセスリクエストに対するバックグラウンドホットコピープロセスの相対的プライオリティが示される。ホットコピープロセスの間、同プロセスの進行状況およびリクエストの種類に応じて、データアクセスリクエストが第一のデバイス11と第二のデバイス12にマップされる。また、ストレージサーバには、ホットコピープロセスにプライオリティを与えるためのリソースが含まれ、ホットコピープロセスのプライオリティが低いと、クライアントプロセッサは、そのデータアクセスリクエストにすぐに対応することができる。ホットコピープロセスのプライオリティが比較的高いと、クライアントプロセッサはそのデータアクセスリクエストへの対応がある程度遅れるが、ホットコピープロセスはより早く完了する。

【０１６６】データセットの転送が完了すると、ステージ3が始まる。ステージ3では、データセットにアドレス

されるクライアントプロセッサからのデータアクセスリクエストが、通信リンク15を通じて第二のデバイス12にルーティングされる。ストレージデバイス11はネットワークから一緒に排除されるか、別の目的で使うことができる。

【０１６７】ストレージサーバ10は、好ましい実施形態において、先に説明したストレージドメインマネージャで構成される。

【０１６８】ストレージデバイス11と12は、独立したデバイスあるいはひとつのストレージユニットにおける論理パーティションからなる。この場合、ホットコピープロセスにより、ストレージユニット内のひとつのアドレスから別のアドレスへとデータが移動する。

【０１６９】図１９、２０、２１、２２は、上述のインテリジェントなネットワークサーバにおいて実行されるホットコピープロセスのソフトウェアをいくつか示したものである。ホットコピープロセスに使用する別のストレージサーバでは、特定のシステムに合わせて構成を超えることができます。仮想回路、持続型テーブルストレージ、ユーザインターフェースの構造に関して、以下の図を見ながら詳細に説明する。

【０１７０】図１９はホットコピープロセスで使用される基本的データ構成を示す。第一の構造350は「ユーティリティ・リクエスト構造体」、第二の構造351は「ユーティリティ構造体」、第三の構造352は「メンバー構造体」と呼ぶ。メンバー構造体352は特定の仮想回路とそのステータスを特定するためのもので、仮想回路の識別子(VDID)、現在仮想回路が取扱っているデータブロックのブロック番号を有する論理ブロックアドレス(LBA)、仮想回路に関するキューにあるリクエスト数、ステータスパラメータ等のパラメータが含まれる。

【０１７１】ユーティリティ構造体351は、現在実行中のユーティリティ、つまりこの場合であればホットコピーユーティリティに関するパラメータを有し、ソースデータセット識別子である「ソースID」、ホットコピープロセス用の1個または複数のデスティネーションストレージデバイスの識別子である「デスティネーションID」、そのユーティリティに関して実行されるリクエストのキュー、現在扱われているブロックとそのサイズを示すパラメータ等のパラメータを記憶する。

【０１７２】ユーティリティリクエスト構造体350は、ホットコピープロセスに関するリクエストを、これに関する各種のパラメータとともに伝える。このパラメータには、例えばリクエストの状態を示すパラメータである「ステータス」、そのリクエストをサポートする各種のフラグ、対応するユーティリティ構造体へのポインタ、クライアントプロセッサからの入力/出力リクエストと比較したそのリクエストのプライオリティを示すパラメータ、ソース内のデータセットを特定するソースマスク、ホットコピープロセスがデータセットをコピーするデスティネーションデバイスの位置を特定するデスティネーションマスク等である。一実施形態において、ひとつのホットコピーリクエストに関する複数のデスティネーションマスクがある。図１９に示すように、ユーティリティリクエスト構造体の中には論理ブロックアドレス(LBA)が保存され、これは現在扱われているデータセット内のデータブロックについて、メンバー構造体の中にも保存される。

【０１７３】ホットコピープロセスを開始するためには、ユーザの入力を受け入れ、これがユーティリティリクエスト構造体を作る。ストレージサーバ内の持続型テーブルストレージはこの構造体で更新され、ソースおよびデスティネーションデバイスのステータスとそのデータに関連する仮想回路がチェックされ、ドライバがホットコピープロセスを開始し、ステータスパラメータが各種データ構造の中にセットされる。ホットコピープロセスの進行状況は、故障時のために持続型テーブルストレージの中に保存される。故障が発生した場合、ホットコピープロセスは、サーバ内の他のリソースおよび持続型テーブルストレージ内に保存されていたステータス情報とデータ構造を使って再開することができる。RAIDモニター等、システム内の他のドライバにはホットコピープロセスが伝えられる。リクエストは、メンバー構造体に入る順番を待つ。

【０１７４】セットアップが完了すると、ホットコピープロセスをサポートする入力、出力プロセスが開始される。このホットコピープロセスをサポートする入力、出力プロセスの相対的プライオリティにより、クライアントプロセッサが同じデータセットに関する入力、出力リクエストを実行している状態で、ホットコピープロセスの進行速度を決定する。好ましいシステムにおいては、クライアントプロセッサからの入力、出力リクエストが最初に実行される。ホットコピープロセスをサポートするブロック転送が実行されている場合、クライアントプロセッサからの入力または出力リクエストを受け取ると、ブロック転送は原子動作として完了し、クライアントプロセッサのリクエストが満たされる。別のシステムでは、プロセスのプライオリティは異なる技術でも管理できる。

【０１７５】ホットコピーを実行する基本的プロセスを図２０に示す。このプロセスは、メンバー構造体のキューの最上位に到達したホットコピーリクエストから始まる(ステップ360)。次にストレージサーバ内のバッファが割り当てられ、ブロック転送をサポートする(ステップ361)。データセット内の第一ブロックのコピーをバッファに移動するメッセージが発行される(ステップ362)。現在のブロックは、ホットコピープロセスについて設定されたプライオリティに従ってバッファに移動される(ステップ363)。ブロックの移動は、ストレージサーバ内で複数のプロセスによるアクセスを制御するための

適当なメモリロックトランザクションを使って行われる。次に、ブロックのコピーをバッファから一つ以上のデスティネーションに移動するメッセージが発行される(ステップ364)。このブロックは、ホットコピープロセスに関するプライオリティに従って、一つ以上のデスティネーションに移動される(ステップ365)。ブロックが移動すると、持続型テーブルストアとプロセスをサポートするローカルデータ構造は、ホットコピーの進行状況を示すステータス情報で更新される(ステップ366)。プロセスは、データセットの最終ブロックがコピーされたか否かを判断する(ステップ367)。コピーが終了していなければ、次のブロックのコピーをバッファに移すメッセージが発行される(ステップ368)。プロセスはステップ363にループし、データセットのブロックを引き続きデスティネーションに移動する。ステップ367において、データセットの最終ブロックがデスティネーションにうまく移動したと判断された場合、プロセスが終了する(ステップ369)。

【0176】本発明の一実施形態によれば、デスティネーションが複数にわたるホットコピープロセスの場合、使用されているデスティネーション群の一つ以上のメンバーがプロセス中に故障することがありうる。この場合、プロセスは動作を続ける一つ以上のデスティネーションで継続することができ、続けられるプロセスをサポートして該当するテーブルの更新が行われる。

【0177】このように、ホットコピー機能は、データセットをまだダウン状態となっていないひとつのメンバーから交換トライブへとコピーするのに使用される。データセットには、ストレージデバイスの内容全体あるいはストレージデバイスの内容の一部が含まれる。ホットコピー機能は、適正にステータスおよびパラメータを管理しながら、どのレベルのRAIDアレイでも使用できる。

【0178】ホットコピーのパラメータには、プロセスのプライオリティ、ソースメンバーデバイス、デスティネーション識別子が含まれる。ホットコピーリクエストには、ソースメンバー識別子、デスティネーションメンバー識別子、コピーブロックのサイズ、コピーの頻度またはプライオリティが含まれる。ホットコピーは、プライオリティに従って、一度にひとつのブロックサイズずつ行われる。現在のブロック位置は、上述のようにデータ構造内のアレイコンフィギュレーションデータの中に保存される。ホットコピープロセスは通常の入力および出力プロセスと同時に実行される。ホットコピーされるドライブへの書き込みは両方のドライブに行われるため、ホットコピーが中断または失敗しても、当初のソースメンバーは有効なままである。ホットコピーが完了すると、当初のソースメンバーはアレイから外され、システムマネジャプログラムによって使用不可と指定される。同様に、一実施形態において、データセットをサポートする仮想デバイスは、新しいデスティネーションを目指すよう更新される。

【0179】図21及び22は、ホットコピープロセス実行中にクライアントプロセッサが発行するデータアクセスリクエストを管理するために、ストレージサーバ内で行われるプロセスを示す。データアクセスリクエストは、読み出しリクエスト、書き込みリクエスト等のうち1種類であっても、同じもののバリエーションであってもよい。その他のリクエストとしては、データチャネル等の管理をサポートするリクエストがある。図21は、書き込みリクエストを扱うひとつのプロセスを示す。

【0180】書き込みリクエストがキューの最上位に到達すると、プロセスが始まる(ステップ380)。プロセスは、この書き込みリクエストが現在のホットコピープロセスの対象となるデータセット内の位置を特定しているかどうかを判断する(ステップ381)。これがホットコピーされているデータセットの中にある場合、プロセスは書き込みリクエストが指示されるブロックがすでにそのデスティネーションにコピーされているかどうかを判断する(ステップ382)。もしコピーされていれば、そのデータセットが最初に保持されていたストレージデバイスと、一つ以上のデスティネーションストレージデバイスの両方に書き込みを行うメッセージが発行される(ステップ383)。次に、入力と出力リクエストのプライオリティに従ってデータが移動され(ステップ384)、プロセスが完了する(ステップ385)。

【0181】ステップ381において、リクエストがデータセットの中にないと、データセットのソースへの書き込みを実行するメッセージが発行され(ステップ386)、この時点でプロセスの流れはステップ384に移る。同様に、ステップ382において書き込み先となる位置がすでにコピーされていることがわかった場合、ソースデバイスに書き込みを行うメッセージが発行される(ステップ386)。

【0182】図22は、ホットコピー中に発生する読み出しリクエストの取扱いを示す。このプロセスは、読み出しリクエストが仮想デバイスに関するキューの最上位に達した時に始まる(ステップ390)。まず、読み出しがホットコピーの対象となるデータセット内であるかどうかが判断され(ステップ391)、読み出しがデータセット内であれば、すでに一つ以上のデスティネーションにコピーされたブロック内であるかどうかが判断される(ステップ392)。読み出しが、デスティネーションにすでにコピーされたブロック内である場合、データを新しい位置から読み出すメッセージが発行される(ステップ393)。別のシステムでは、システム内のデータトラフィックの管理に影響する信頼性、スピードその他の要因に応じて、ソースデバイス、あるいはソースおよびデスティネーションデバイスの両方から読み出しが行われる。ステップ393以降、データはクライアントプロセッサのデータアクセスリクエストに関するプライオリティに従っ

てリクエスタに返される(ステップ394)。ここで、プロセスは終了する(ステップ395)。

【0183】ステップ391において、読み出しリクエストがホットコピーの対象となるデータセット内にはないと判断された場合、ソースデバイスを読み出すメッセージが発行される(ステップ396)。同様に、ステップ392において、読み出しリクエストがまだデスティネーションにコピーされていないブロックにアドレスされていると判断された場合、ソースデバイスからデータを読み出すメッセージが発行される(ステップ396)。ステップ396以降、プロセスはステップ394に戻る。

【0184】ブロックがストレージサーババッファを通じて移動されている間に、特定のブロック内でデータの読み出しまたは書き込みリクエストが発生した場合、このリクエストの扱いを管理するにはデータロックアルゴリズムが使用される。例えば、読み出しまたは書き込みリクエストを受け取っている間にホットコピープロセスをサポートして論理ブロックがロックされると、クライアントプロセッサは、データがロックされているために、この読み出しまたは書き込みリクエストが拒絶されたという通知を受け取る。クライアントプロセッサに高いプライオリティを与える別のシステムにおいて、読み出しまたは書き込みリクエストは続けられ、その一方でホットコピーをサポートするバッファの中に保持されていたブロックは削除され、ホットコピーのステータスがリセットされて、そのブロックが移動されていないことが示される。個々の利用に関する必要に応じて、各種のデータロックアルゴリズムを利用できる。

【0185】[ターゲットエミュレーション]図1乃至3に示す構成において、ストレージサーバはデータのユーザとデータを保存するストレージドメインにおけるストレージデバイスとの間の中間デバイスとして動作する。この環境では、レガシーストレージデバイス、つまりサーバを中間デバイスとして挿入する前にあったデバイスをサポートするために、サーバにはレガシーストレージデバイスをエミュレートするリソースが供給される。このように、サーバがレガシーデバイスとデータのユーザとの間に挿入されると、サーバは、ユーザとレガシーデバイスの間で使用されているストレージチャネルプロトコルに従ってレガシーデバイスの論理アドレスを仮想的に決定する。次にストレージサーバは、受け取ったレガシーデバイスにアドレスされたすべてのリクエストをそのプロトコルに従って処理する。さらに、必要な構成情報をレガシーデバイスから再生し、ローカルメモリにこの情報を保存して、レガシーデバイスにおいてユーザが予想するステータスおよび構成情報が、サーバ内のローカルリソースを使って供給されるようにする。これにより、サーバとレガシーシステムの間の通信を省くことができ、サーバはストレージチャネルプロトコルに従ってレガシーデバイスの動作をスプーフし、ストレージネットワークにサーバを追加する際、ユーザの再構成が不要となる、あるいは大幅に簡略化される。

【0186】[まとめ]ストレージエリアネットワーキング(SAN)は、新しいストレージ中心コンピューティングアーキテクチャである。主に光ファイバチャネルベースのストレージサブシステムとネットワークコンポーネントが利用可能となったことにより、SANは高速データアクセスとデータ移動、よりフレキシブルな物理的構成、ストレージ容量の利用改善、中央集中化されたストレージ管理、オンラインストレージリソースの利用と再構成、ヘテロジニアスな環境を約束する。

【0187】旧来の「ダイレクトアタッチストレージ」モデルにおいて、ストレージリソースはひとつのサーバだけに通じる高速直接物理パスを有し、他のサーバはすべて、LANを通じて間接的にのみ、そのストレージリソースに極めて低速でアクセスしていた。ストレージエリアネットワークは、「ネットワークされた」トポロジーにおいて個々のサーバから個々のストレージリソースに直接高速アクセスパスを提供することにより(光ファイバチャネルを使用)、これを変えている。ネットワークアーキテクチャの導入もまた、ストレージ構成のフレキシビリティを大幅に向上させ、特定のサーバからストレージリソースを分離し、サーバサイドのリソースにほとんど影響を与えずにこれらを管理、構成することを可能にしている。

【0188】SANは今日の環境におけるフレキシビリティとデータアクセスのニーズに応えるための正しいトポロジーを提供する一方で、SANのトポロジーそのものは十分にビジネス上の問題に対処しているとはいえない。単にスイッチ、ハブ、ルータ等のSANファブリックコンポーネントを通じてサーバとストレージリソース間を物理的に接続するだけでは、SANの可能性を十分に実現することはできないが、SANファブリックが、必要とされるセキュリティが保たれた中央集中的なストレージ管理機能を実現するためのハードウェアインフラストラクチャを提供していることは確かである。これら2つの開発を組み合わせて利用することで、新しい環境におけるビジネス上の目標を達成するために不可欠なフレキシビリティが得られ、重要データにいつでもどこでもアクセス可能となる。

【0189】SANハードウェアのインフラストラクチャの上に必要となる管理機能はストレージドメイン管理である。最適なストレージのフレキシビリティと高性能のアクセスを実現するために、ストレージドメイン管理はサーバやストレージデバイスではなく、SANそのものの中に設置するのが最も効率的である。サーバベースおよびストレージベースのリソースを用いるアプローチは、サーバサイドでもストレージサイドでも異機種混合状態を十分にサポートできないため、最適とはいえない。

【0190】ストレージドメイン管理は、既存のSANハ

ードウェアインフラストラクチャ上に設置される中央集中化され、セキュリティの保たれた管理機能であり、ヘテロジニアスな環境への高性能、高アベイラビリティの高度なストレージ管理能力を提供する。ストレージドメイン管理の目的は、従来の機器と新しい機器とを統合し、サーバおよびストレージリソースをSANとストレージ管理タスクから解放し、すべてのSANコンポーネントを通じて利用できるSANベースのアプリケーションをホストすることのできる堅牢なSANファブリックの中核を構成することである。SANはストレージドメイン管理を使用せずに構築できるが、最適化されたSAN環境を構築、管理するには、この極めて重要な間能力が必要となる。

【0191】ストレージドメイン管理の基本要素には以下のものがある:

- 異機種間の相互運用性
- セキュリティの保たれた中央集中管理
- スケーラビリティとすぐれた性能
- 企業クラスの信頼性、可用性、保守性
- 特定用途に作成されたインテリジェントなプラットフォーム

ストレージドメイン管理の分野により、顧客はSANの全能力を活用してビジネス上の問題に対処することが可能となる。

【0192】サーバとストレージの連結や今日の新たな事業状況で一般的となった合併買収により、ヘテロジニアスな環境に対応できることは企業環境において死活問題である。単独メーカーの製品ラインのためにSAN機能を提供するような製品セットでは、顧客はSANの全能力を実現できず、新たなサーバやストレージ製品を追加してこれを利用してもなお、旧来の機器への投資を保持する必要があるため、ストレージドメインマネジャは最低でも光ファイバチャネルとSCSIアタッチメントをサポートでなければならない。ストレージドメインマネジャはそのうちに導入される新たな技術に適用できるよう進化していく必要があるため、プラットフォームは確実に成長し、より広範なマルチプロトコルの接続性が実現する。SANは中央集中的に管理することのできる大型の仮想化されたストレージプールを作るため、特にバックアップ/再生、災害復旧において、従来の「ダイレクトアタッチ」ストレージアーキテクチャと比較して、ストレージ管理作業が縮小される。SANはすべてのサーバからすべてのストレージへの物理的アクセスパスを効果的に提供するものの、ストレージがすべて論理的にすべてのサーバにアクセスできるとは限らないため、セキュリティの問題には確実な方法で対処しなければならない。SANファブリックメーカーはこれを、「ゾーン」を論理的に定義することによって行っており、各サーバがそのゾーン内にあると定義されたデータにしかアクセスできないようになっている。明らかに、安全なゾーンあるいはストレージ「ドメイン」を定義する能力はストレージドメインマネジャのひとつの要素である。ポートレベルではなくLUNレベルでゾーン内に含まれるものを定義する等、ドメインをより細かく定義することにより、今後ストレージアセットの利用をさらに柔軟に改善することができる。

【0193】ストレージドメインマネジャは、メーカーを問わず、接続されたすべてのサーバとストレージを通じてひとつの管理インターフェースから利用できる、総合的な中央集中化されたストレージ管理機能を提供する。中央から、システムアドミニストレータはヘテロジニアスなストレージリソース間のデータの移動またはミラーリングを管理し、長期にわたり、さまざまなヘテロジニアスなストレージリソースに対してこれらの機能を動的に利用することができる。その結果、大幅なコスト削減と管理の簡素化が実現する。拡張可能なインテリジェントなプラットフォームとして、ストレージドメインマネジャは完全な中心位置に設置され、接続されたサーバとストレージリソースすべてにわたって利用できるストレージ管理機能をホストする。

【0194】新たな事業状況によって拍車がかかるストレージの拡張率からみると、あるSAN環境について、ストレージ容量はそのライフタイムの中で簡単に100倍にも膨れ上がる。SANの中央知能として位置付けられるストレージドメインマネジャも、負荷に対応して性能が劣ることがないよう、急激な成長に適応できなければならない。広い動作範囲について、スムーズでコスト効率のよいスケーラビリティを実現するために、構成の拡張とともに知能も付け加えて行くべきである。インテリジェントなプラットフォームにおける大量のデータをキャッシュメモリに保存する能力により、SAN構成は最適化され、特定用途型の環境における性能が向上する。例えば、ファイルシステムジャーナルやデータベーステーブルインデックスまたはログといった「ホットスポット」がストレージドメインマネジャそのものの中にある高速ストレージの中にキャッシュされると、ストレージドメインマネジャを使わずに構築されたより旧式のSAN構成と比較して、メッセージパスの待ち時間が大幅に短縮される。オンボードのストレージが大量であることを考えると、データベースとファイルシステム全体が効果的にキャッシュされ、性能の大幅改善が実現する。オンボードストレージ能力はまた、データ移動およびその他のデータ移行作業中にデータをステージ分けする上でも重要である。前述のように、SANへ移行する主な理由のひとつは、全体としてのデータへのアクセス可能性を改善することである。この新しいストレージアーキテクチャに移行した結果として故障箇所が1ヵ所でも発生すると、これによる利点の多くは実現することができない。このため、データそのものだけでなく、そのデータまでのアクセスパスも常に利用できる状態になければならない。故

障によるダウンタイムは、自動I/Oパスフェイルオーバ、論理ホットスペアリングおよびプラグ接続可能、ホットスワップ可能なコンポーネント等、相対的な内部コンポーネントや機能を使用することによって短縮しなければならない。ダウンタイムは、オンラインファームウェアのアップグレード、ハードウェアとソフトウェアの動的再構成、高性能なバックグラウンドデータ移動等のオンライン管理機能を使ってさらに縮小する必要がある。

【０１９５】最高レベルの性能を確保するために、ストレージドメインマネジャは、特にそれが必要とされるストレージ関連タスクのために最適化された、特定用途向けに作られたインテリジェントなプラットフォームとするのが好ましい。このプラットフォームは、データ移動とストレージ管理アプリケーションの実行に必要なローカルでの高速ストレージにより裏打ちされ、さまざまなストレージ管理タスクを実行する重大なローカル処理能力をサポートする。

【０１９６】インテリジェントなストレージサーバとして汎用プラットフォームを使用する場合と比較して、特定用途向けに作られたプラットフォームは、はるかに高速でより決定的な応答時間を実現するリアルタイムオペレーティングシステム、メッセージの待ち時間を短縮する、より効率的なI/Oパスコード、アプリケーションエンジンではなくデータムーバエンジンとして最適化されたオペレーティングシステムカーネルを提供する。この特定用途向けに作成されたプラットフォームは、汎用オペレーティングシステムでは得られない、信頼性の高い画期的なメッセージ送信等のカーネルレベルの機能をサポートする。統合パスフェイルオーバ、オンライン管理および動的再構成等の高アベイラビリティという特徴は、中核オペレーティングシステムによってサポートされる。ヘテロジニアスなSAN環境をサポートするために最適な場所に知能を持たせることにより、ストレージドメインマネジャはエンドユーザに以下のような業務上の利点をもたらす:

- ストレージアセットの割り当てと利用の改善
- 急成長する流動的なストレージ環境にコスト効率よく対応するフレキシビリティ
- オンライン管理と構成を通じた高いアベイラビリティ
- ストレージ管理の全体的$/ギガビットコストを下げる、より効率的な管理
- 統合SAN環境において各種のサーバとストレージをひとつにまとめる能力
- 全ストレージリソースを通じて動的に利用できるストレージ管理とキャッシング機能を追加することによってJBODストレージの価値を高める。

【０１９７】ストレージドメイン管理と同時に採用される堅牢なSANハードウェアインフラストラクチャは、急速かつ予測不能な変化を続ける環境に対応しながらも、アベイラビリティの高いデータに確実かつ高速にアクセスするというフレキシビリティを提供する。このように実現される中央集中的ストレージ管理パラダイムは、企業にとって他社との競争上の利点を提供する、より効率的で低コストのデータ拡張管理方法である。

【０１９８】本発明の各種実施形態に関する上の記述は、例を挙げ、説明するためのものであり、本発明を説明中紹介した具体的形態のみに限定しようとするものではない。さまざまな変更や同等の配置は、当業者には自明である。

【０１９９】

【発明の効果】以上説明しだように、本発明によれば、SANアーキテクチャのフレキイビリティ及び能力を活用しつつストレージシステムの管理を簡素化するシステムを提供できる。

【図面の簡単な説明】

【図１】（ａ）は、本発明によるストレージサーバをストレージドメイン管理のストレージルータまたはストレージディレクタとして構成したものを含むストレージエリアネットワークを示す図、（ｂ）はインテリジェントなストレージエリアネットワークサーバのいくつかの用途を示す図

【図２】本発明によるストレージサーバをヘテロジニアスなネットワークにおけるストレージドメイン管理のストレージルータまたはストレージディレクタとして構成したものを有する別の構成によるストレージエリアネットワークを示す図

【図３】本発明による複数のストレージサーバを相互に直接通信チャンネルを持たせて構成し、より広範なストレージドメインまたは複数のストレージドメインをサポートするようにした、より複雑なストレージエリアネットワークを示す図

【図４】本発明によるストレージドメイン管理をサポートするストレージサーバのブロック図

【図５】本発明によるストレージドメイン管理をサポートするストレージサーバの別の例を示すブロック図

【図６】インテリジェントなストレージエリアネットワークサーバのハードウェアアーキテクチャのブロック図

【図７】インテリジェントなストレージエリアネットワークサーバ用オペレーティングシステムのソフトウェアモジュールおよびサポートプログラムのブロック図

【図８】本発明によるシステムで用いる光ファイバチャネルインターフェース用ハードウェアドライバモジュールの簡略図

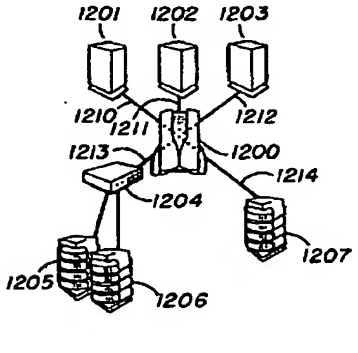
【図９】本発明によるハードウェアドライバモジュールを含むソリッドステートストレージシステムの簡略図

【図１０】本発明によるストレージサーバの一実施形態において用いるディスクドライブの内部アレイの図

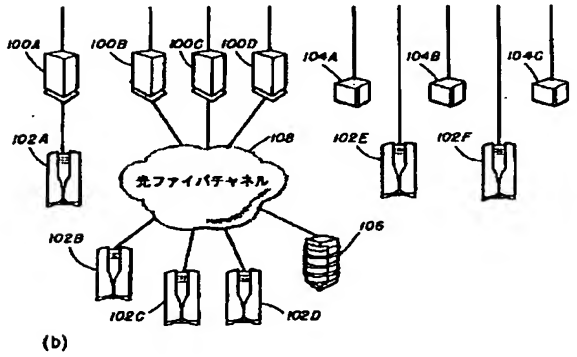
【図１１】ローカルアンサー機能を備えた本発明によるターゲットサーバ内部サービスモジュールの概略図

【図１２】ディスクミラーを利用した内部サービスモジュールの図
【図１３】パーティション機能を利用した内部サービスモジュールの図
【図１４】キャッシュ機能を利用した内部サービスモジュールの図
【図１５】本発明による仮想回路構成を示す図
【図１６】本発明による持続型テーブルストアマネジャを利用した内部サービスモジュールの図
【図１７】本発明による持続型ストレージハードウェアドライバモジュールの概略図
【図１８】本発明による、３段階のホットコピーリソースを有する中間デバイスを備えたネットワークの簡略図
【図１９】本発明によるホットコピープロセスを用いたドライバの一例において用いられるデータ構成を示す図
【図２０】本発明によるドライバによって実行されるホットコピープロセスを示すフローチャート
【図２１】ホットコピープロセス中の書き込みリクエストの取扱いを示すフローチャート
【図２２】ホットコピープロセス中の読み出しリクエストの取扱いを示すフローチャート
【符号の説明】
1201, 1202, 1203…クライアントサーバ
1204…ハブ
1205, 1206, 1207…デバイス
1210, 1211, 1212…クライアントインタフェース
1213, 1214…ストレージインターフェース

【図１】

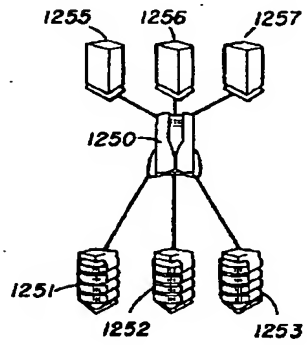


(a)

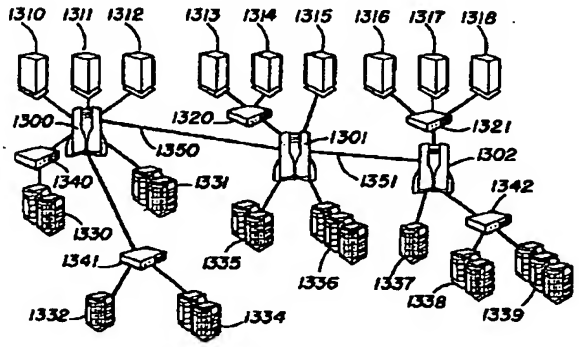


(b)

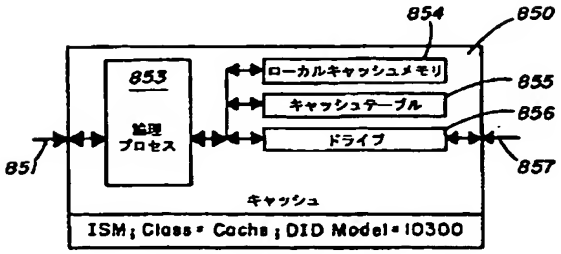
【図２】



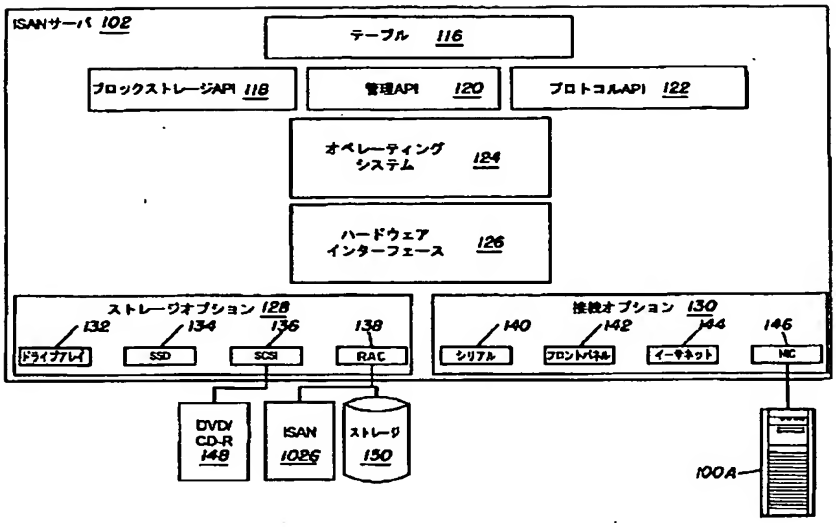
【図３】



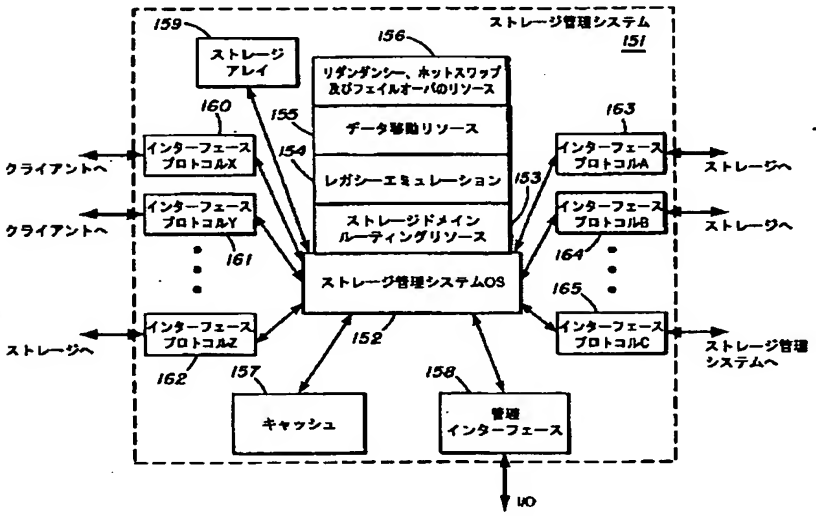
【図１４】



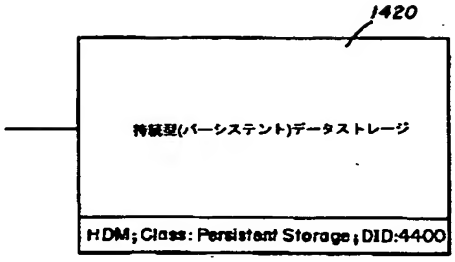
【図4】



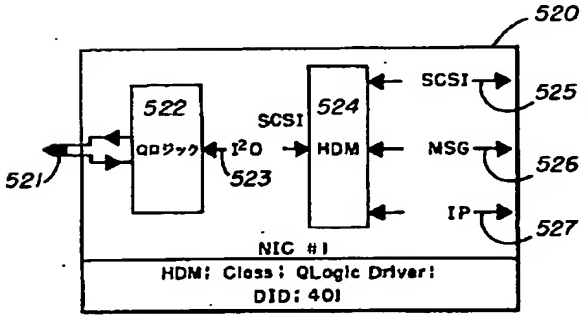
【図5】



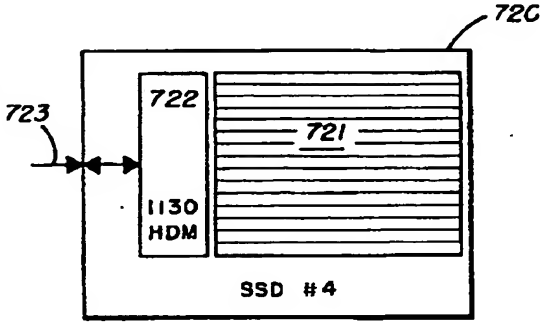
【図17】



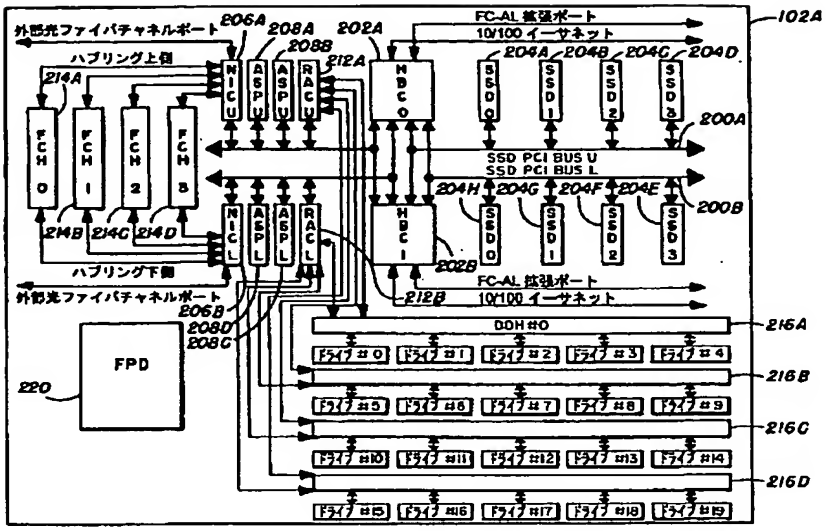
【図8】



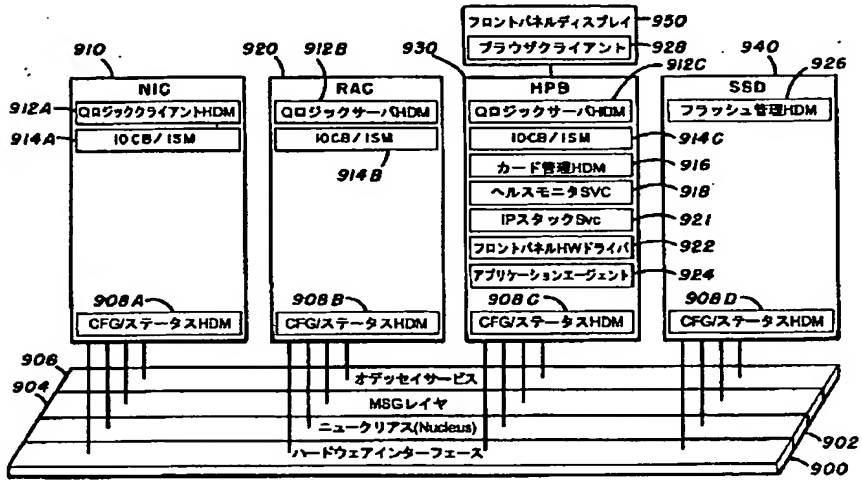
【図9】



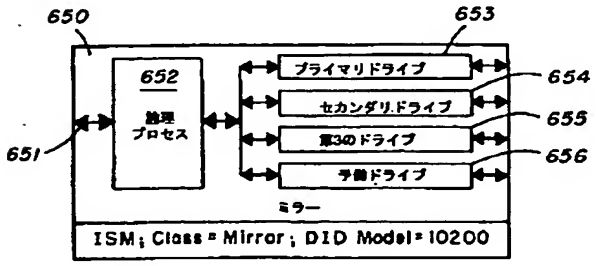
【図6】


外部光ファイバチャネルポート
206A
208A
202A
FC-AL 拡張ポート
10/100 イーサネット
102A
ハブリング上側
212A
204A
204B
204C
204D
200A
SSD PCI BUS U
SSD PCI BUS L
204H
204G
204F
204E
200B
ハブリング下側
212B
202B
206B
208D
208C
216A
216B
216C
216D
FPD
220

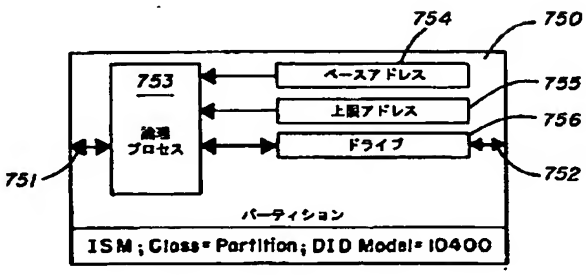
【図7】


フロントパネルディスプレイ
950
ブラウザクライアント
928
910
920
912B
930
940
926
912C
NIC
RAC
HPB
SSD
912A
914A
914B
914C
916
918
921
922
924
908A
908B
908C
908D
906
904
902
900
オデッセイサービス
MSGレイヤ
ニュークリアス(Nucleus)
ハードウェアインターフェース

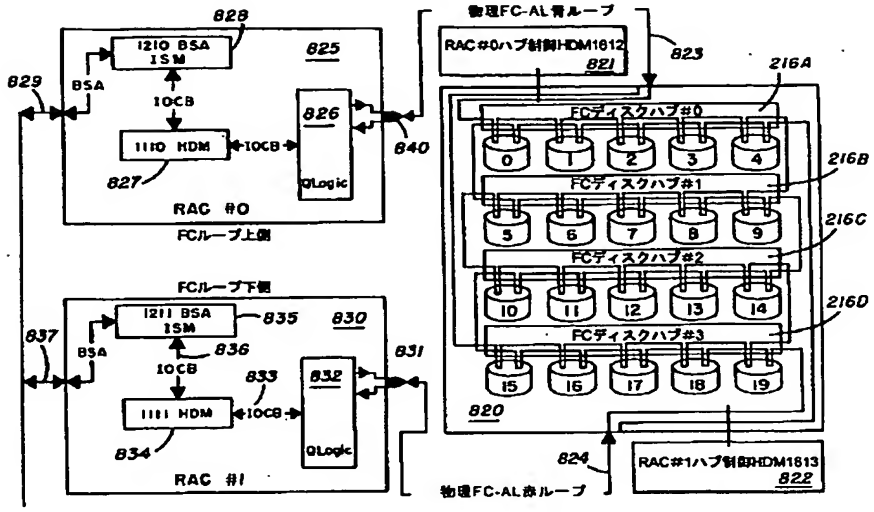
【図12】


650
653
652
プライマリドライブ
654
セカンダリドライブ
655
第3のドライブ
656
予備ドライブ
651
ミラー
ISM; Class = Mirror; DID Model = 10200

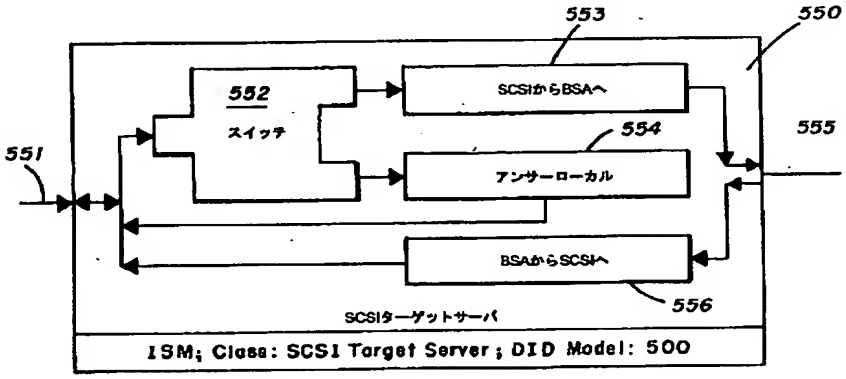
【図13】


754
750
753
ベースアドレス
755
上限アドレス
756
ドライブ
751
752
パーティション
ISM; Class = Partition; DID Model = 10400

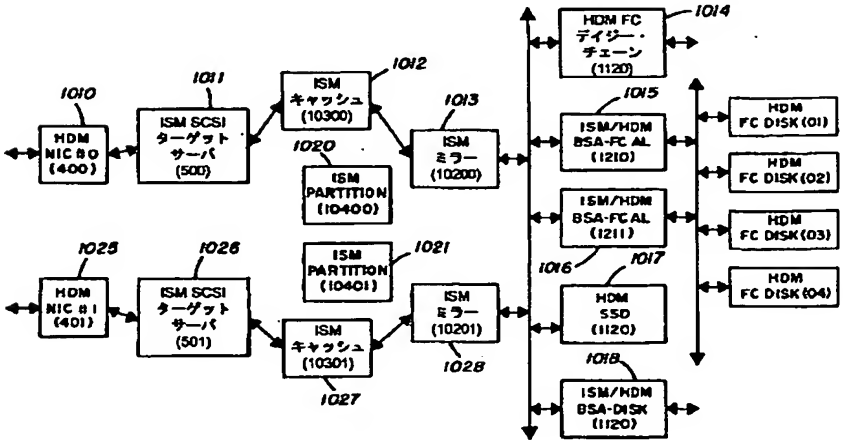
【図10】


828
1210 BSA ISM
825
829
BSA
IOCB
826
1110 HDM
IOCB
827
QLogic
RAC #0
FCループ上側
物理FC-AL青ループ
RAC#0ハブ制御HDM1812
821
823
216A
FCディスクハブ#0
0 1 2 3 4
216B
FCディスクハブ#1
5 6 7 8 9
216C
FCディスクハブ#2
10 11 12 13 14
216D
FCディスクハブ#3
15 16 17 18 19
820
840
FCループ下側
1211 BSA ISM
835
830
837
BSA
836
IOCB
833
832
831
1111 HDM
IOCB
834
QLogic
RAC #1
824
RAC#1ハブ制御HDM1813
822
物理FC-AL赤ループ

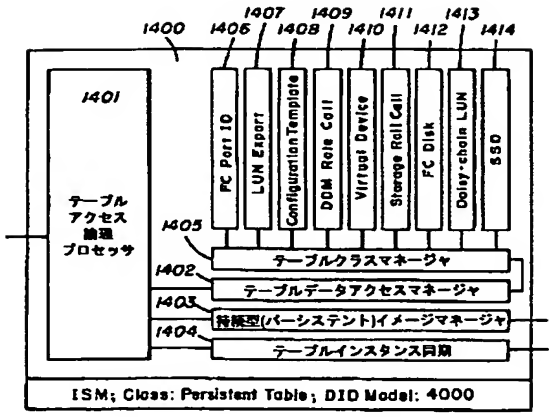
【図11】


553
550
552
スイッチ
SCSIからBSAへ
554
555
551
アンサーローカル
BSAからSCSIへ
556
SCSIターゲットサーバ
ISM; Class: SCSI Target Server ; DID Model: 500

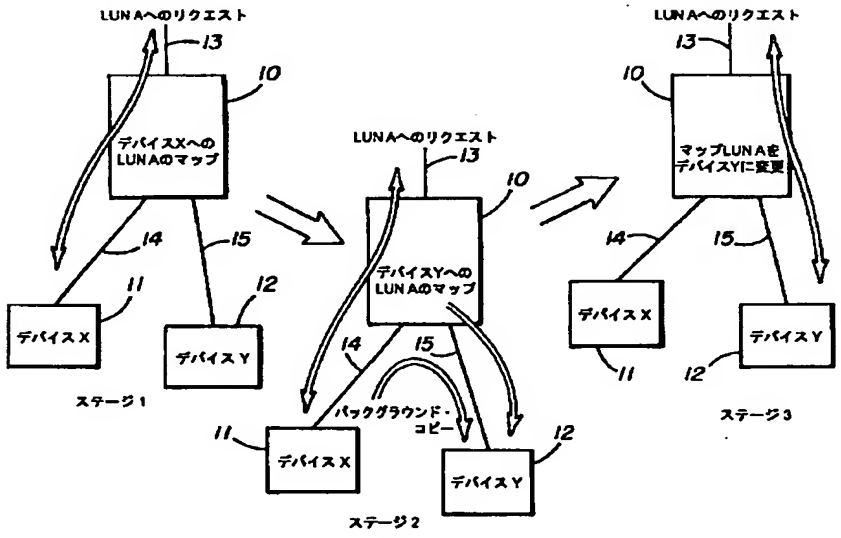
【図15】


1014
HDM FC デイジー・チェーン (1120)
1010
HDM NIC #0 (400)
1011
ISM SCSI ターゲット サーバ (500)
1012
ISM キャッシュ (10300)
1013
ISM ミラー (10200)
1020
ISM PARTITION (10400)
1015
ISM/HDM BSA-FC AL (1210)
HDM FC DISK(01)
HDM FC DISK(02)
ISM/HDM BSA-FC AL (1211)
HDM FC DISK(03)
1016
1017
HDM SSD (1120)
HDM FC DISK(04)
1025
HDM NIC #1 (401)
1026
ISM SCSI ターゲット サーバ (501)
ISM PARTITION (10401)
1021
ISM キャッシュ (10301)
ISM ミラー (10201)
1028
1027
1018
ISM/HDM BSA-DISK (1120)

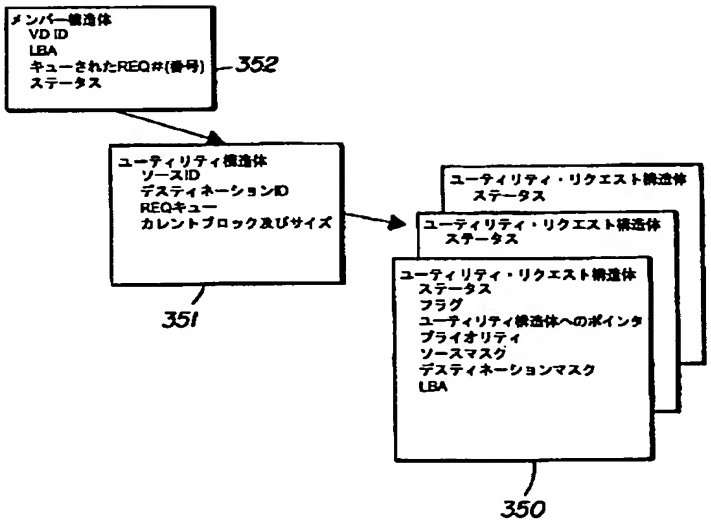
【図16】


1400
1401
テーブル
アクセス
論理
プロセッサ
1405
1402
1403
1404
1406
1407
1408
1409
1410
1411
1412
1413
1414
PC Port ID
LUN Export
Configuration Template
DDM Role Call
Virtual Device
Storage Roll Call
FC Disk
Daisy-chain LUN
SSD
テーブルクラスマネージャ
テーブルデータアクセスマネージャ
特殊型(パーシステント)イメージマネージャ
テーブルインスタンス同期
ISM; Class: Persistent Table; DID Model: 4000

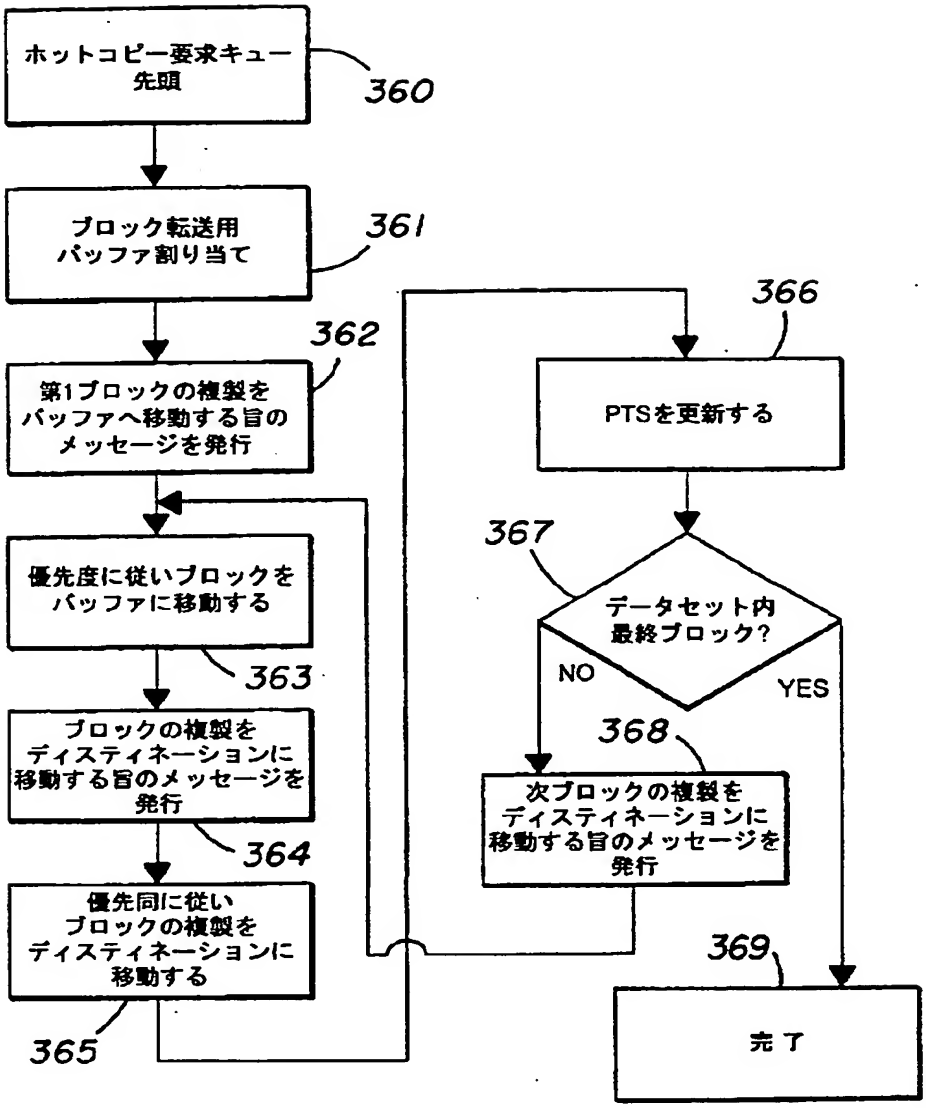
【図18】


LUNAへのリクエスト
13
10
デバイスXへの
LUNAのマップ
14
15
11
12
デバイスX
デバイスY
ステージ1
LUNAへのリクエスト
13
10
デバイスYへの
LUNAのマップ
14
15
バックグラウンド・
コピー
11
12
デバイスX
デバイスY
ステージ2
LUNAへのリクエスト
13
10
マップLUNAを
デバイスYに変更
14
15
デバイスX
デバイスY
11
12
ステージ3

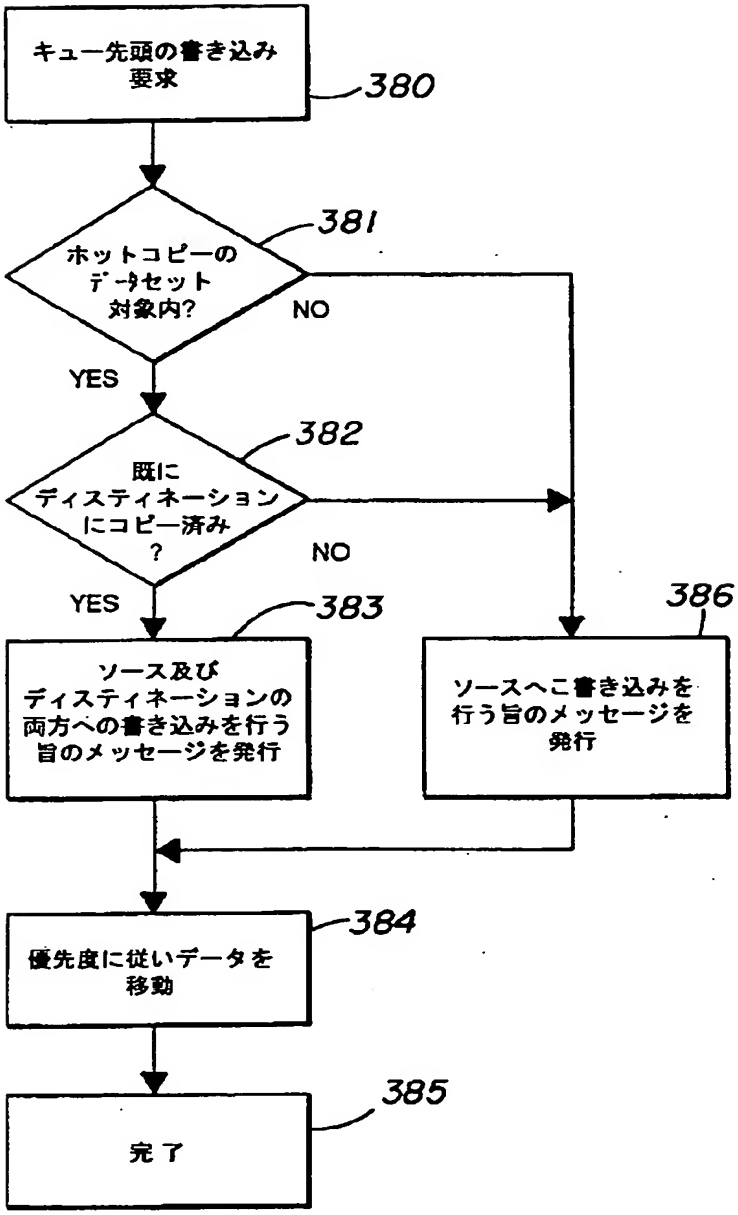
【図19】


メンバー構造体
VD ID
LBA
キューされたREQ#(番号)
ステータス
352
ユーティリティ構造体
ソースID
デスティネーションID
REQキュー
カレントブロック及びサイズ
351
ユーティリティ・リクエスト構造体
ステータス
ユーティリティ・リクエスト構造体
ステータス
ユーティリティ・リクエスト構造体
ステータス
フラグ
ユーティリティ構造体へのポインタ
プライオリティ
ソースマスク
デスティネーションマスク
LBA
350

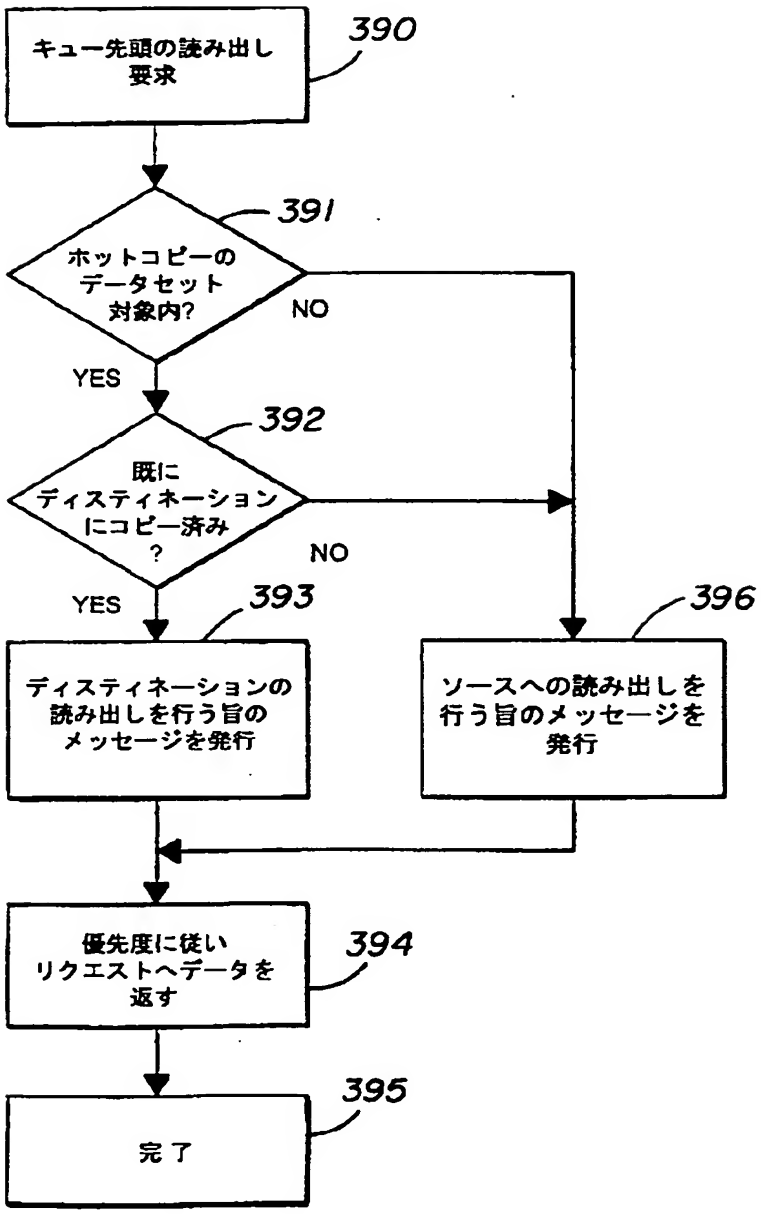
【図20】


ホットコピー要求キュー先頭
360
ブロック転送用バッファ割り当て
361
第1ブロックの複製をバッファへ移動する旨のメッセージを発行
362
優先度に従いブロックをバッファに移動する
363
ブロックの複製をディスティネーションに移動する旨のメッセージを発行
364
優先同に従いブロックの複製をディスティネーションに移動する
365
PTSを更新する
366
データセット内最終ブロック?
367
NO
YES
次ブロックの複製をディスティネーションに移動する旨のメッセージを発行
368
完了
369

【図21】


キュー先頭の書き込み要求
380
ホットコピーのデータセット対象内?
381
NO
YES
既にディスティネーションにコピー済み?
382
NO
YES
ソース及びディスティネーションの両方への書き込みを行う旨のメッセージを発行
383
ソースへこ書き込みを行う旨のメッセージを発行
386
優先度に従いデータを移動
384
完了
385

【図22】



フロントページの続き

(31)優先権主張番号 455106
(32)優先日 平成11年12月6日(1999. 12. 6)
(33)優先権主張国 米国(US)
(31)優先権主張番号 482213
(32)優先日 平成12年1月12日(2000. 1. 12)
(33)優先権主張国 米国(US)

(71)出願人 597001637
One Dell Way, Round Rock, TX 78682-2244, United States of America

(72)発明者 アラン・アール・メレル
アメリカ合衆国、カリフォルニア州 94539、フレモント、チェニン・ブランク・ドライブ 48835
(72)発明者 ジョセフ・アルトマイヤー
アメリカ合衆国、アイオワ州 52327、リバーサイド、ファイブハンドレッドフォーティース・ストリート・エスダブリュ、3689
(72)発明者 ジェリー・パーカー・レーン
アメリカ合衆国、カリフォルニア州 95136、サン・ジョセ、トニノ・ドライブ 4829
(72)発明者 ジェームス・エー・テイラー
アメリカ合衆国、カリフォルニア州 94550、リバーモア、フロレンス・ロード 1033
(72)発明者 ロナルド・エル・パークス
アメリカ合衆国、カリフォルニア州 94526、ダンビル、ムスタング・コート 55
(72)発明者 アラステアー・テイラー
アメリカ合衆国、カリフォルニア州 95123、サン・ジョセ、カレロ・アベニュー 755
(72)発明者 シャリ・ジェイ・ノラン
アメリカ合衆国、カリフォルニア州 95132、サン・ジョセ、ピナクル・ドライブ 3470
(72)発明者 ジェフリー・エス・ネスボー
アメリカ合衆国、カリフォルニア州 94566、プリーザントン、コルテ・ベラ・クルズ 2720
(72)発明者 ジョージ・ダブリュ・ハリス・ジュニア
アメリカ合衆国、カリフォルニア州 94041、マウンテン・ビュー、ビュー・ストリート 327
(72)発明者 リチャード・エー・ルグォー・ジュニア
アメリカ合衆国、ニュー・ハンプシャー州 03051、ハドソン、パインウッド・ロード 11